SHARP

INTERFACE FOR PC-1500

MODEL CE-158

INSTRUCTION MANUAL

CONTENTS

INTRODUCTION

Congratulations on your purchase of this SHARP CE-158 Interface. The CE-158 is an optional device for use with the PC-1500 Pocket Computer and incorporates an RS-232C interface and a parallel interface. For the parallel interface, refer to the description on and after page 65.

SPECIAL NOTICE TO SHARP CUSTOMERS

1. Your CE-158 is equipped with a rechargeable Ni-Cad battery, so that you may obtain maximum efficiency from your unit, please read the instructions below before operating the unit for the first time.

    After purchasing, or when stored unused for over three months, we recommend that you recharge the battery before using the unit.
    Simply connect the AC adaptor (EA-21A only) between the unit and the wall outlet. When recharging this unit, turn the power switch OFF. (See diagram on page 72.) The batteries are fully charged after 15 hours.

OPERATIONAL NOTES

To insure trouble-free operation of your CE-158 interface, we ask you to observe the following instructions:
1. This interface should be kept in areas free from extreme temperature changes, moisture and dust.
2. A soft, dry cloth should be used to clean the interface. Do not use any solvent or wet cloth.
3. If service of your interface is required, consult only an authorized SHARP service center.
4. Keep this manual for further reference.

WHAT IS AN RS-232C INTERFACE?

The term "RS-232C" represents the specific standard of the EIA (Electronic Industries Association in U.S.A.).
This standard specifies the interfaces for use in data transmission between the modem and communication control devices, and the transmission of digital binary series data, control signals and timing signals between the modem and data terminal equipment.
The RS-232C interface is one of devices for general use in information exchanges between data processing equipment and data terminal equipment.

The SHARP CE-158 (RS-232C/interface) is designed to comply with the EIA standards. However, we cannot guarantee that the CE-158 will work with all so-called "RS-232C compatible" devices.
Nor do we commit ourselves to provide engineering and programming support for such applications or any other special custom-use situation.
For actual use of the CE-158, read this instruction manual carefully and understand the data terminal equipment.
This instruction manual contains the information needed for information exchange with data processing equipment.

Serial interface

The serial interface is used to make communication (data transfer) with information carried on a single signal line.

Serial signals carried on the signal line of the interface should either be handled in the start/stop or asynchronous mode. Parallel signals are first converted into serial signals before being transmitted, and the signals received are then converted from serial to parallel.

In addition to the data signal line, there are a number of signal lines required for exchanging operating status on sending and receiving sides, though they vary depending on the device used.

The serial interface will offer a wide range of applications when used in conjunction with the number of peripheral units that are equipped with the RS-232C interface, such as:

- Acoustic couplers that perform data communication via the telephone network,
- Printer,
- X-Y plotter,
- Instrument device,
- Personal computer,
- Video display unit.

4

CONNECTION OF PC-1500 WITH CE-158

CE-158

Power switch

PC-1500

Power-off key

Power-on key

Joint plate (A)
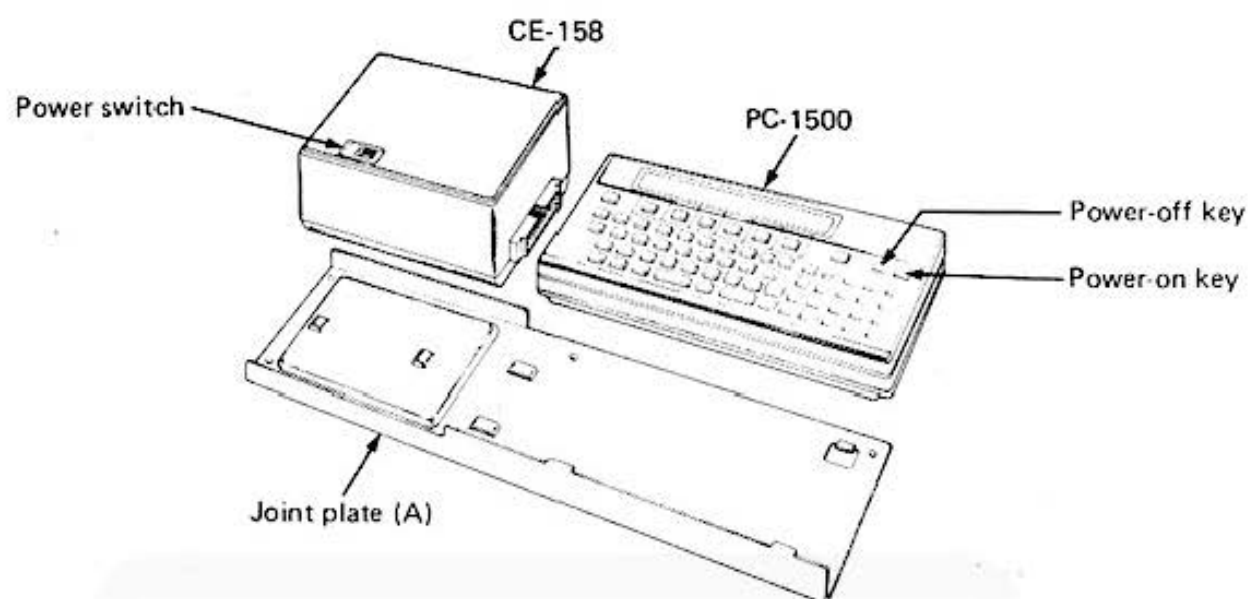
CONNECTION OF PC-1500 WITH CE-150 AND CE-158

PC-1500

CE-150

CE-158

Connector for
parallel interface

AC adaptor
connection jack

Connector
for RS-232C

Joint plate (B)

Screw

5
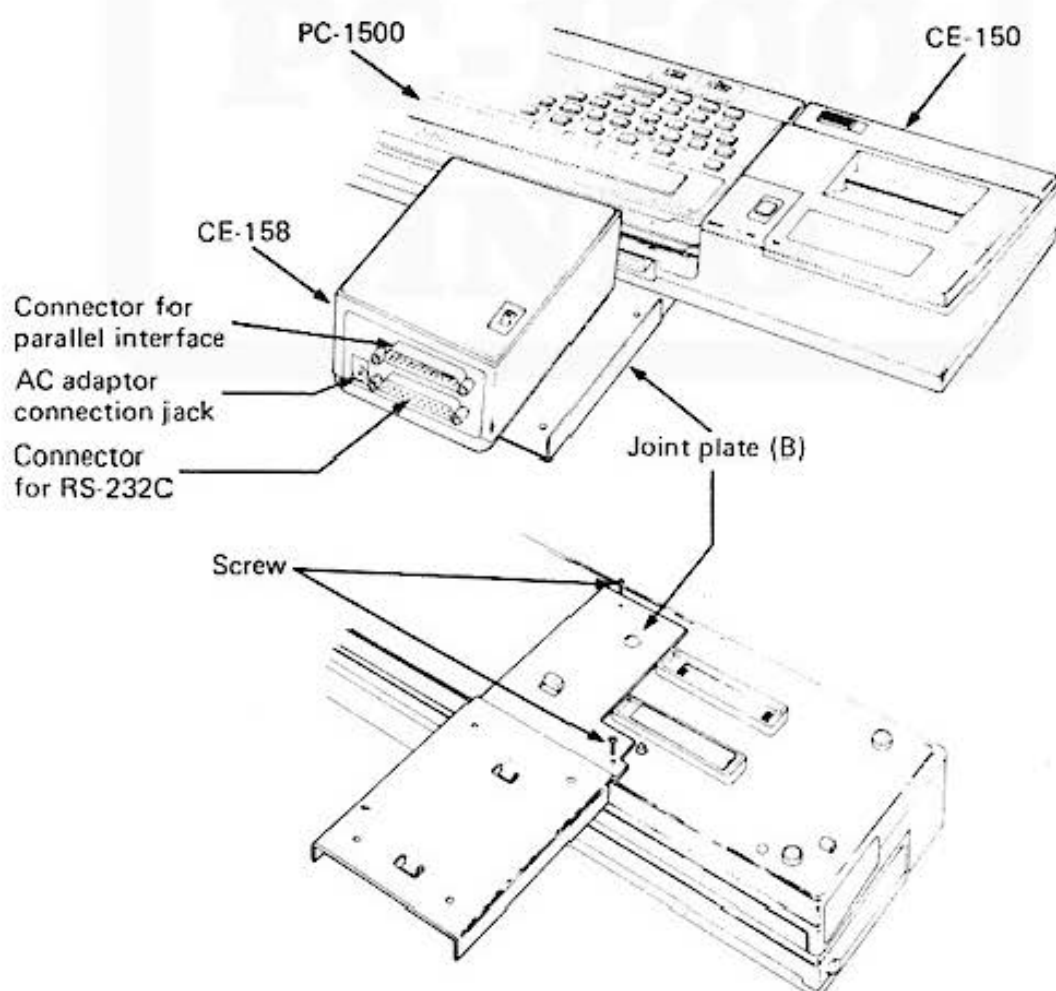
RS-232C INTERFACE

RS-232C INTERFACE SPECIFICATIONS

Transmission method	:	Asynchronous
Applicable standards	:	EIA RS-232C compliance
Baud rate	:	50, 100, 110, 200, 300, 600, 1200, 2400 baud, programmable *
Data bit	:	5, 6, 7, 8 bits, programmable
Parity bit	:	Even, odd, non-parity, programmable
Stop bit	:	1 for the character size of 5 to 8.
		1.5 for the character size of 5. } programmable
		2.0 for the character sizes of 6 to 8.
Connectors used	:	60-pin male connector for connection with the PC-1500 or CE-150.
		25-pin connector, DB-25(W), for connection with an external device.
		Adaptor jack.
Power supply source	:	4.8 V ⎓ (DC): Ni-Cad rechargeable battery
		AC: Local voltage with EA-21A
Power consumption	:	4.8 V ⎓ (DC), 0.80 W
AC adaptor/charger	:	EA-21A Local voltage
Battery capacity	:	For approx. 3 hours of operation (charging: 15 hours)
Output signal level	:	High level: +5 V to +10 V (3 to 7 Kohms load)
		Low level: −5 V to −10 V (3 to 7 Kohms load)
Interfacing signals	:	Inputs: RD, DSR, CD, CTS
		Outputs: TD, RTS, DTR
		Others: SG (,FG)
Switch	:	x1 (POWER switch)
Dimensions	:	86 (W) x 115 (D) x 50 (H) mm
		3-3/8″ (W) x 4-17/32″ (D) x 1-31/32″ (H)
Weight	:	435 g (0.96 lbs.)
Accessories	:	Keyboard templates, joint plates (two kinds) and instruction manual.

* : In terminal program mode, the specifications of baud rate (600, 1200 and 2400) is restricted.

COMMAND REFERENCE TABLE

[BASIC program mode]

● **Statements**
SETCOM, SETDEV, OUTSTAT
INPUT, INPUT$, INPUT%, INPUT#
PRINT, PRINT#
LPRINT, LLIST
CLOAD, CLOADa, CLOADr
MERGE, MERGEa
CSAVE, CSAVEa, CSAVEr
FEED, ZONE, CONSOLE, TRANSMIT
PRINT#−8, INPUT#−8,

● **Functions**
COM$, DEV$, INSTAT
RINKEY$
SPACE$
ERN, ERL

[Terminal program mode entry instruction]
TERMINAL
DTE

7

O : Yes, X : No

Command	Example	Manual execution			Program execution	Function
		RUN	PRO	RESERVE		
[Statement]						
SETCOM	SETCOM300,7,E,1	O	O	O	O	Sets communication parameters.
	SETCOM	O	O	O	O	Initializes communication parameters.
SETDEV	SETDEV KI,PO	O	O	O	O	Assigns communication device.
	SETDEV	O	O	O	O	Releases communication device.
OUTSTAT	OUTSTAT 0	O	O	O	O	Sets status of the RS-232C port output signals.
INPUT	INPUT "A=";A	X	X	X	O	Substitutes the variable with the input data thru the RS-232C port.
INPUT$	INPUT$ "A=";A	X	X	X	O	Same as above.
INPUT%	INPUT% A$(*)	O	O	O	O	Same as above.
INPUT#	INPUT# A	O	O	O	O	Same as above.
PRINT	PRINT 123	O	O	O	O	Sends data thru the RS-232C port.
PRINT#	PRINT# A	O	O	O	O	Same as above.
LPRINT	LPRINT 123	O	O	O	O	Same as above.
LLIST	LLIST 10,100	O	O	O	O	Sends program thru the RS-232C port.
CLOAD	CLOAD "FILE"	O	O	O	X	Loads program thru the RS-232C port.
CLOADa	CLOADa	O	O	X	X	Loads program thru the RS-232C port in ASCII code.
CLOADr	CLOADr	O	O	O	X	Loads reserve program thru the RS-232C port.
CSAVE	CSAVE "FILE"	O	O	O	O	Saves program or reserve program thru the RS-232C port.
CSAVEa	CSAVEa	O	O	O	O	Saves program thru the RS-232C port in ASCII code.
CSAVEr	CSAVEr	O	O	O	O	Saves reserve program thru the RS-232C port.
FEED	FEED 5	O	O	O	O	Sends the END code.
ZONE	ZONE 20	O	O	O	O	Specifies block length during data output by LPRINT statement.
CONSOLE	CONSOLE 80,0	O	O	O	O	Specifies digits/line and the kind of END code.

Command	Example	RUN	PRO	RESERVE	Program execution	Function
		Manual execution				
TRANSMIT	TRANSMIT BREAK, 10	O	O	O	O	Sends long spaces (padding).
MERGE	MERGE	O	O	X	X	Links program with the program sent thru the RS-232C port.
MERGEa	MERGEa	O	O	X	X	Links program with the program sent thru the RS-232C port in ASCII code.
PRINT#−8,	PRINT#−8, 123	O	O	O	O	Sends data out thru the RS-232C port.
INPUT#−8,	INPUT#−8, A	O	O	O	O	Substitutes the variable with the data sent through the RS-232C port.
[Function]						
COM$	COM$	O	O	O	O	Character string with which communication parameters are set.
DEV$	DEV$	O	O	O	O	Character string with which communication device is set.
INSTAT	INSTAT	O	O	O	O	Designates RS-232C port handshake signal state.
RINKEY$	RINKEY$	O	O	O	O	Character string with which last one byte code input thru the RS-232C port immediately before execution of command is set.
SPACE$	SPACE$ (10)	O	O	O	O	Character string consisting of specified number of spaces.
ERN	ERN	O	O	O	O	Designates error code.
ERL	ERL	O	O	O	O	Designates error line number.
[Terminal program entry instruction]						
TERMINAL	TERMINAL	O	O	O	O	Default Protocol: XON/XOFF ON, ECHO OFF
DTE	DTE	O	O	O	O	Default Protocol: XON/XOFF OFF, ECHO ON Parameter: 300 Baud 7 bits data Even Parity 1 Stop bit

9

Serial signal

Transmit and receive data are handled in a form of serial signal as illustrated below. (Waveform as observed at the I/O pin of the DB-25W.)
Absence of data must be stop bits.
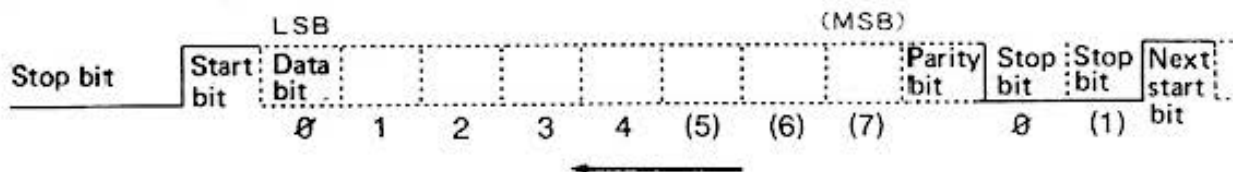
Data bits, 5, 6, 7, parity bit, and stop bit 1 may not exist, depending on the communication parameter used.

(1) Transmission speed (baud rate)
   Represents number of bits transmitted per second, including the start bit, data bit, parity bit, and stop bit.

(2) Character length
   Represents number of bits used to comprise a single transmission character.

(3) Parity bit
   A bit attached next to the last data bit (MSB). Three modes of non-parity, even parity, and odd parity can be chosen to check data accuracy.

(4) Stop bit(s)
   Bit(s) attached next to the data bit or parity bit; the size varies from 1 to 1.5, and 2.

Interfacing signal

Signal levels at the I/O pins of the DB-25W connector shall be as follows:

Input signal
    High level (SPACE or ON state of signal): +(3 to 15) V
    Low level (MARK or OFF state of signal): −(3 to 15) V

Output signal
    High level: +(5 to 10) V
    Low level: −(5 to 10) V

As the level of the output signal is measured with the load of 3 to 7 Kohms with the cable length of 1 meter. Please note that there may be a case when the above condition is not met, if the load is out of above range or the cable length is longer, or the cable capacity is larger.

10

RS-232C

The RS-232C is the standards of the EIA (Electronics Industries Association) which has been established to set the standard interfacing requirements between the data terminal unit and the data com munication unit.

The CE-158 has been developed to meet those standards.

Main parts of these standards define the following conditions:

(1) Electrical characteristics
(2) Mechanical characteristics of the interface unit
(3) Functions of interfacing signals

(1) Electrical characteristics

1) Input signal below −3 V is regarded as MARK, and above +3 V as SPACE.
2) Load side impedance shall be a direct current resistance of more than 3 Kohms against an impressed voltage of less than 25 V, or less than 7 Kohms against an impressed voltage of 3 to 25 V.
3) Output signal in MARK state shall be −5 to −15 V or output signal in SPACE state shall be +5 V to +15 V.
4) Effective load capacitance at the end shall be less than 2500 pF, including cable capacitance.

(2) Functions of interfacing signals

Connector signal configuration
The DB-25(W) connector comes with the CE-158.
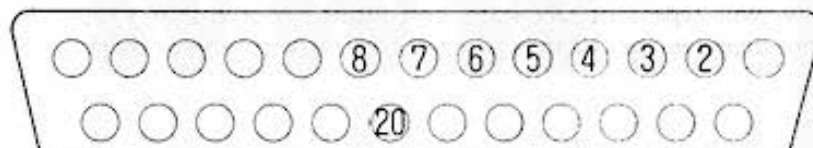
Connector Pin Configuration

Actually, eight pins are used out of twenty five pins.

NOTE: Different signal connections may be required depending on signals used by the device connected.

Use the metal case (type DB24659).

11
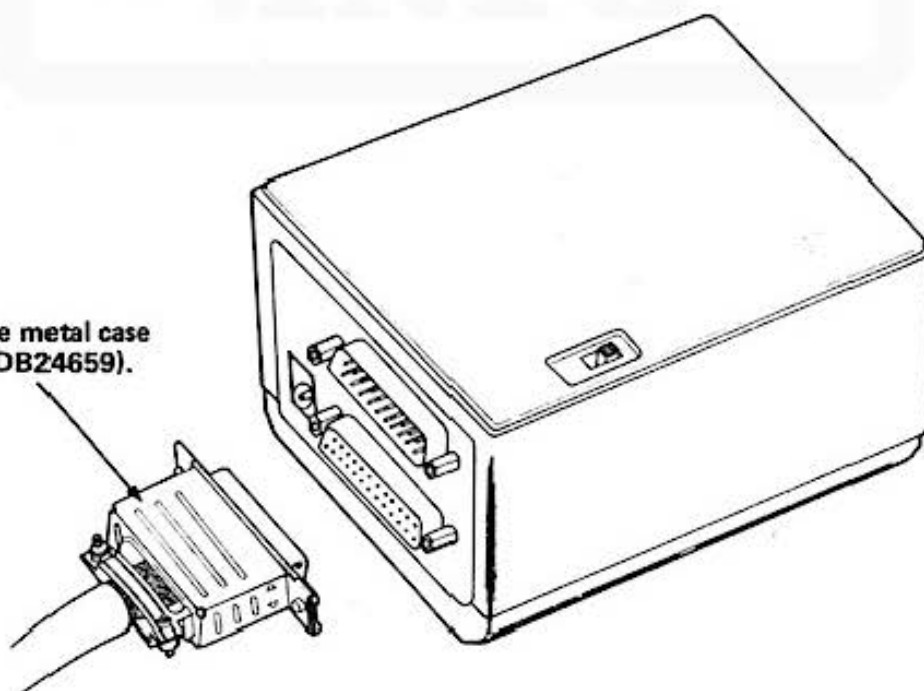
Pin assignment

Pin No.	Item	Symbol	In/Out	Function	Remarks
2	Transmit Data	TD	Out	Send data ±(5–10) VDC signal	
3	Receive Data	RD	In	Receive data ±(5–15) VDC signal	
4	Request-To-Send	RTS	Out	Modem transmission carrier control. ON (V+): Send carrier. OFF (V−): Stop carrier.	OFF in Power On.
5	Clear-To-Send	CTS	In	Data transmission control. ON: Data transmission enabled. OFF: Data transmission disabled.	OFF when a Cable isn't connected.
6	Data Set Ready	DSR	In	Modem status indicator. ON: Modem ready to send/receive. OFF: Modem not ready to send/receive.	ON when a cable isn't connected.
7	Signal Ground	SG		Provides reference voltage level with respect to all interfacing signal.	
8	Carrier Detect	CD	In	Carrier detection. ON: Carrier signal received. OFF: Carrier signal not received.	ON when a cable isn't connected.
20	Data Terminal Ready	DTR	Out	Status indication on terminal side (CE-158). ON: Terminal ready. OFF: Terminal not ready.	OFF in Power On.

NOTE: Input signal within a range of +5 to +15 V and output signal within a range of +5 to +10 V will represent ON state, and input signal within a range of −5 to −15 V and output signal within a range of −5 to −10 V will represent OFF state.

12

INTERFACING CABLE

CAUTION: The illustrations are for typical equipment and are used as examples only. Your particular equipment manual should be consulted for specific cable connections.

Connection with acoustic coupler

Connect cable signal lines in the following manner when the unit is connected with an acoustic coupler.

CE-158 side (DB-25M)　　　　　　　　　　Accoustic coupler side (DB-25M)

	CE-158			Coupler	
2	T D		2	T D	
3	R D		3	R D	
4	R T S		4	R T S	
5	C T S		5	C T S	
6	D S R		6	D S R	
7	S G		7	S G	
8	C D		8	C D	

Connection with Personal Computer

Connect cable signal lines in a manner described below, when the unit is connected with Personal Computer.

CE-158 side (DB-25M)　　　　　　　　　　Personal Computer side (DB-25M)

	CE-158			Personal Computer	
2	T D		2	T D	
3	R D		3	R D	
4	R T S		4	R T S	
5	C T S		5	C T S	
6	D S R		6	D S R	
7	S G		7	S G	
8	C D		8	C D	
20	D T R		20	D T R	

13

To make connection with other peripheral units, check specifications of the peripheral unit if they should match with those of the CE-158, and use an appropriate cable to connect with.

Cautions described below are basic requirements and they are exempt in terms of software. You will have to judge for yourself after reading the command description.

(1) Electrical characteristics
Make sure that it is in compliance with the EIA RS-232C standards.

(2) Transmission method
The CE-158 is operated in the asynchronous mode, not the synchronous mode.

(3) Transmission procedure
No procedure is used.

(4) Baud rate and transmission data format
Make sure that it is in compliance with baud rate and data format.

(5) Code system
This unit uses the ASCII code or internal codes in use.

(6) Connector
Use the 25-pin DB-25(W) connector.

(7) Interfacing signal
As four input signal lines and three output signal lines are used, it will require four lines for signal input. In case one output line should be connected with two input lines, attention must be paid to the load resistance as it may not meet with the EIA requirement.

(8) Signal flow to peripheral
Because this unit is a data terminal, connector signals are designed for transmission between the data terminal and the data communication unit. Therefore, it needs to use the cable where signal flow is inverted, if the peripheral unit is also a data terminal. (See figure below.)
The majority of data communication units are acoustic couplers, but the printer, X-Y plotter, personal computer, etc., are data terminal in every case.

2	TD		TD	2
3	RD		RD	3
4	RTS		RTS	4
5	CTS		CTS	5
6	DSR		DSR	6
7	SG		SG	7
8	CD		CD	8
20	DTR		DTR	20

CE-158   Peripheral
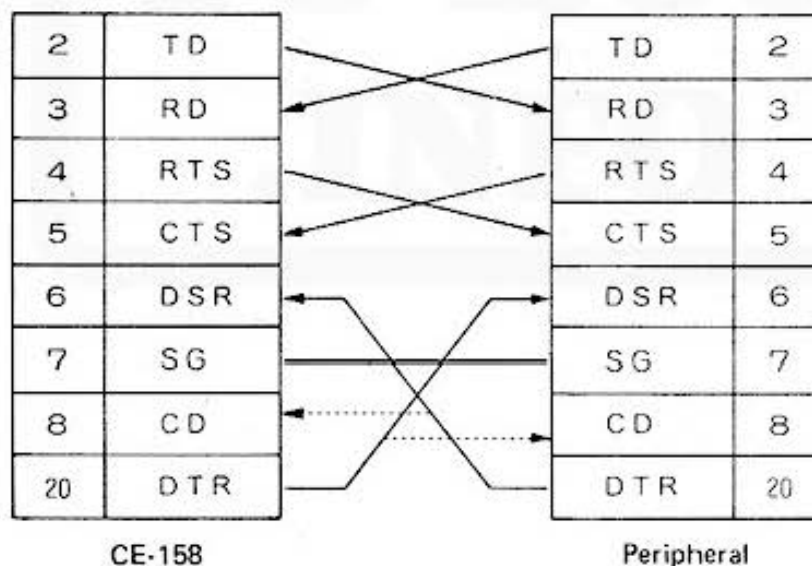
• So far as CD and DSR concerned, it will be regarded as the ON state when open, while other inputs are regarded as the OFF state.
• When DTR is connected to CD in addition to DSR, it causes both input signal levels to go out of given ratings (+3 — 15 V and −3 — 15 V) dissatisfying the EIA requirements.

14

FUNCTIONS

This unit has the following functions:

1. BASIC program mode

2. Terminal program mode

The mode 1 controls program and data transaction using the BASIC program commands to perform the following applications:
- Data exchange with personal computer,
- Listing of program contents and operational result on printer.
- Processing of measurement data from instrumentation.

On the other hand, the mode 2 functions as a terminal unit without need of preparing program, for instance:
- As the required item is specified to the data base through the keyboard, the relevant information is fetched on the display or printed on the printer.
- Used for the output unit of a peripheral unit that incorporates no output feature.

When the CE-158 is connected with the PC-1500 or CE-150, be sure to turn off the computer (PC-1500).
As the CE-158 is controlled by command statements and functions used for data transmission via the RS-232C port, data transmission with the peripheral unit must be controlled by means of those command statements.
Statements and functions pertinent to this unit hold execution unless RS-232C port input signals, CTS, CD, and DSR, are all active. (Execution will take place as soon as all of these signal go active.) Therefore, the state of the peripheral unit must be checked using the INSTAT function before execution of the statement.
Because the status output signals to the peripheral unit, RTS and DTR, are inactive after power on, they have to be turned active using the OUTSTAT statement before commencement of data transmission.

IMPORTANT:
This unit has to be connected with the PC-1500 when program is created using statements and functions pertinent to the CE-158. Otherwise, it will not insure proper operation at the time of program execution.
Also, it needs connection with the PC-1500 when the program that uses statements and functions pertinent to the CE-158 is executed or listed.

1. Low battery and POWER switch
The CE-158 is operated by the internal Ni-Cad batteries (AA x 4).
(1) When the PC-1500 POWER switch is turned on after turning the CE-158 POWER switch on, power is supplied to the CE-158 from these Ni-Cad batteries.
(2) When the PC-1500 POWER switch is turned off in above condition (1), it turns off the battery power supply to the CE-158.
(3) Turn off the CE-158 POWER switch to save unnecessary power consumption when the external device is not operating.

(NOTE) To execute the instruction of the CE-158, be sure to turn on the CE-158 power switch. When the instruction is executed with the CE-158 power off, the ERROR 50 will occur. To change the program, the CE-158 power switch may be in any position.

15

The following messagge may be put on the display of the PC-1500 when the CE-158 POWER switch is off before turning on the PC-1500 POWER switch or when power supply from Ni-Cad batteries is low.

```
: C H E C K    8
```

In this case, recharge the battery or turn on the CE-158 POWER switch.
When the CE-158 is connected with the CE-150 and the battery voltage of the CE-150 is also low,

```
: C H E C K   6 8
```

is displayed.

Under the following condition,

```
N E W Ø ? : C H E C K   8 ·
```

may be displayed. However, NEW Ø [ENTER] operation is not required in this case.

- When the PC-1500 is turned ON under the Auto Power Off condition
- And CE-158 power is OFF or the battery of CE-158 is low.

In this case, turn on the CE-158 POWER switch or recharge the battery, and press the [CL] key to clean the PC-1500.

(NOTE)   Do not connect or disconnect the AC adaptor to the CE-158, when the program is executed with the PC-1500. Otherwise, the ERROR 50 may occur.

Supply voltage of PC-1500

The CE-158 will not perform a proper operation when the battery indicator " · " looks weak as seen from the front, as it alerts low battery condition for the PC-1500. The battery indicator is located to the right side of the display. In such a case, either replace the battery of the PC-1500, with the fresh one or connect the AC adaptor EA-150 into the adaptor jack of the PC-1500.
The EA-150 is an AC adaptor supplied with for the Printer/Cassette Interface CE-150.

2. Initial state after power on after [OFF] key power off.

When power is turned on to the PC-1500, the CE-158 is placed in the following initial state.

Item	Parameter	Description
SETCOM	300, 8, N, 1	
SETDEV		All devices are cleared.
OUTSTAT	3	DTR and RTS = OFF
CONSOLE	0,0	END code = CR code
ZONE	13	
TAB	0	

Although power-on after auto-power-off resumes the same condition immediately before auto-power-off, it may be in the state of "OUTSTAT 3" until a command is executed for the CE-158.

16

Signal state at power-off

(1) Output signals from the CE-158 such as TD, RTS, and DTR, become unstable when the following condition is met while sending or receiving programs or data via the CE-158.

    (a) Power on and off of the PC-1500.

    (b) Manipulation of the POWER switch of the CE-158.

    (c) Releasing the low battery condition of the CE-158 after recharge.

    (d) Manipulation of the AC adaptor EA-21A.

If the opposite side is in receiving operation at this point, it may read unstable data as though data is received. In such an event, observe one of the following methods.

- Start the execution all over again (transfer of program, for example).
- Send the header together when the PC-1500 sends the data out so that the receiving side will receive the data that follows to the header. (Example will be in Figure below.)
- Send the same contents of the data for three times when the PC-1500 sends data out. So that the receiving side will receive the similar data for more than two times.

(2) Note that the opposite side may sometime assume the CE-158 output signal (RTS, DTR) to be ON state when the PC-1500 or CE-158 is in one of following conditions.

    (e) PC-1500 in the power off state.

    (f) The POWER switch of the CE-158 is off.

    (g) The CE-158 in low battery condition.

    (h) The CE-158 is not connected with the PC-1500 or CE-150 that is connected with the PC-1500.

(Example): Shown below is a program example that the Personal Computer processes the data sent from the PC-1500, provided that the Personal Computer is in receiving condition at all times.

PC-1500 + CE-158		Personal Computer
10 SETCOM 300.8.E.1		
20 SETDEV PO.KI		
30 OUTSTAT 0		100 RSMODE A.R8. T8.M71.RX1
		300 ON ERROR GOTO 1000
	Send of header "ABC" →	310 RSI A A$···Receive of header
100 LPRINT "ABC"		320 FOR D=1 TO 500 : NEXT D
110 INPUT B$		330 IF RIGHT$(A$.3) "ABC"
120 IF B$ "Y"THEN100		THEN RSO A "N" :GOTO310
130 SETDEV PO		···· Header confirmed
		340 RSO A "Y"
200 LPRINT A$	Send of data "A$" →	400 RSI A A$
		900 END
900 END	Processing of received data	
	Error process routine	1000 FOR D=1 TO 500:NEXT D
	[for receiving error or incorrect header]	1010 RSMODE A.R8. T8.M71.RX1
		1020 RSO A "N"
		1030 GOTO 300

17

BASIC PROGRAM MODE

1. SETCOM (statement)

Used to set communication parameters such as Baud Rate (BR), Word Length (WL), Parity (PR) and Stop (ST) bits.
The following initial parameters will be set upon power on.

(300, 8, N, 1)

Stop bit = 1

Parity = non-parity

Word length = 8

Baud rate = 300
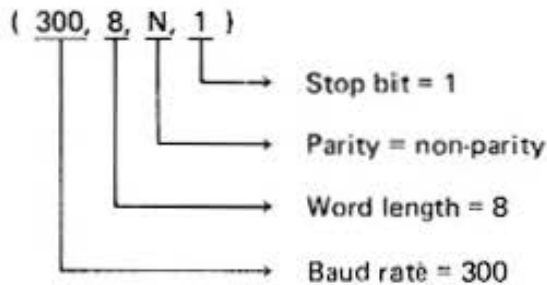
(1) SETCOM

If no parameters were specified, above initial parameters will be the default.

(2) SETCOM ──→ BR,WL,PR,ST ──→

──→ Non-numeric ──
      variable

BR : 50, 100, 110, 200, 300, 600, 1200, 2400
WL : 5, 6, 7, 8
PR : N (Non), E (Even), O (Odd)
ST : 1
    2 (actual ST = 1.5 when WL = 5)

● Omission of parameter entry will assign the previously used parameters.

(EX)   10   SETCOM   : (300,8,N,1)

          20   SETCOM 1200, ,E,2   : (1200,8,E,2)

          30   AB$="300,7,E,1"

          40   SETCOM AB$: (300,7,E,1)

NOTE: When the contents of the nonnumeric variable is null, it will assume the same condition as (1).
When parameter designation is suspended in the middle of entry, parameters after suspension will remain the same as previously assigned.

(EX)   SETCOM 600

2. COM$

Nonnumeric function used to get communication parameters set.

(EX)   At power on:

300,8,N,1

Upon execution of SETCOM 1200,7,E,1:

1200,7,E,1

3. SETDEV

With this command is determined whether the INPUT, PRINT, LPRINT, LLIST, CLOAD, INPUT#, MERGE, CSAVE, or PRINT# command should execute the I/O assignment to the RS-232C port (setup of assignment direction).

Assignment direction must be indicated by the assignment of KI, DO, PO, CI or CO after SETDEV.

The command furnished with the assignment direction performs I/O operation through the RS-232C port. If not furnished, it executes the command of the PC-1500 or CE-150.
All assignments will be cleared after power on.

Assignment	I/O	RS-232C execution command when set
KI	I	INPUT
DO	O	PRINT
PO	O	LPRINT, LLIST
CI	I	CLOAD, INPUT# (,MERGE)
CO	O	CSAVE, PRINT#

SETDEV ───→ KI,DO,PO,CI,CO ───┬──→
         └─→ Non-numeric variable ─┘
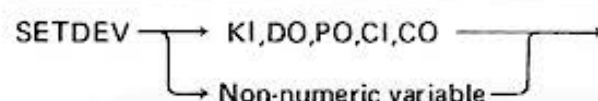
There is no restriction for the sequence of assignments.

NOTE-1: Once the SETDEV statement has been executed, all assignments before that will become invalid.
To clear all assignments, either omit assignment or reset the contents of the non-numeric variable.

NOTE-2: Clear all assignment to use TAB of the CE-150, because it will result in ERROR 27 when; for instance, "TAB expression" is executed with assignment set by the SETDEV statement.

4. DEV$

Nonnumeric function used to get present assignment direction set.

(EX) At power on:

Null

Upon execution of SETDEV DO,CI:

DO,CI

5. OUTSTAT

Used to set the RS-232C handshake signal (RTS, DTR) conforming to low order two bits of the specified expression ($-32768 \sim 32767$).
Handshake signal can be defined as follows:

$2^{15} \sim 2^5$	Nothing	I/O
2^4	Data Set Ready (DSR)	In
2^3	Carrier Detect (CD)	In
2^2	Clear To Send (CTS)	In
2^1	Request To Send (RTS)	Out
2^0	Data Terminal Ready (DTR)	Out

Bit = { 0 : SPACE (= high level, V+ at connector)
       { 1 : MARK (= low level, V- at connector)

RTS and DTR are 1 at power on.

6. INSTAT

Function used to get RS-232C port handshake signal state (0 ~ 31).
NOTE: Execution of the RS-232C port I/O command execution will be suspended when a bit of any of input signals, DSR, CD, or CTS is "1".
Execution will take place as soon as it turns to "0".

7. INPUT

Effective only when KI has been declared by the SETDEV statement.
● ASCII code data sent through the RS-232C port is substituted with the variable.

(1) INPUT —→ variable ———→
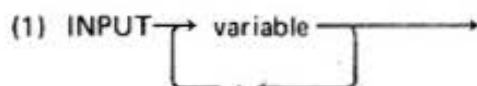
Data is received through the RS-232C port with a question mark "?" on the LCD display.

(2) INPUT ——→ "GUIDANCE", —→ variable ———→
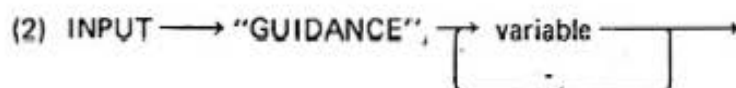
Same as (1) except that "GUIDANCE" is put on the LCD display.

(3) INPUT ——→ "GUIDANCE"; —→ variable ———→
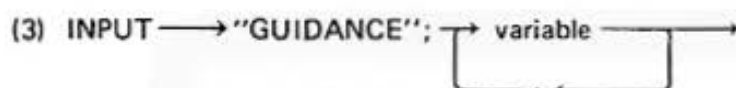
Same as (2).

Data will be received through the RS-232C port until the CR code is met. (Maximum size of data will be 80 digits.)
Existence of a comma (,) before the CR code, will be assumed to be a data separator.
If the number of the variable specified by the INPUT statement is smaller than that of the input data, it will result in ERROR 65. If larger, on the contrary, data are received for as many as specified in the statement, and the last data of the input is put on the display. For (1) and (2) above, the question mark "?" and the message "GUIDANCE" are put out of the display, but for (3), the last data part is put on the display after the message "GUIDANCE."

NOTE-1:   In the "INPUT", "INPUTS" and "INPUT-8," statements, the "(double quotation) and , (comma) can not be substituted with nonnumeric variables.
If the above (" or ,) is used, the ERROR 65 will be caused.

NOTE-2:   Data ending with the CR code is translated into intermediate computer language if the variable to be substituted next is a numeric variable.
If substituting variables continue to be numeric variables, nonnumeric variables and so on, with the received data separated into several sections by a comma, it may result in error when there is a code string in the data (i.e. RUN, PRINT, SIN, etc.) which is to be substituted with the nonnumeric variable.

NOTE-3:   If the first data from the RS-232C port is the CR code, that program line is skipped.

(EX)   10   OUTSTAT 0

20   SETCOM 300 , 8 , N , 1

30   SETDEV KI

40   INPUT "IN EXCURSION OF KI", A , B$, C

50   LPRINT A ; B$; C

60   END

Key operation	Display (or printout)	Serial input data
RUN [ENTER]	IN EXCURSION OF KI (DEG) ″ ″ (DEG) SIN30 (3RS-232C_0.5) (DEG) ⟩	1+2. RS-232C CR SIN30 CR

Cautions during execution of INPUT and INPUT# statement

Before entering data by executing program on the PC-1500 (see figure below), it will be neces-sary that the data sending side should confirm that the PC-1500 is in the receiving condition (execution of 30th line) before starting the data.

```
10  SETDEV KI
20  OUTSTAT 0
30  INPUT A, B
```

To send data to the PC-1500, the sending side should have a wait time before transmitting data. Since the wait time may vary depending on the contents of the input data or the program, it has to have sufficient allowance.

You will be able to create the wait time of about "INT X/64 sec" by means of the statement below, if the transmit side is the PC-1500.

SETDEV: WAIT X: PRINT

Depression of the PAPER FEED key when the CE·150 is in connection will not enable to receive data, because the PC-1500 halts its operation as long as paper is fed.

Note : Create the wait time over
"20/baud rate" (second) before the PC-1500 send the data to the RS-232C port and executes the OUTSTAT 1 ~ 3.

Example: SETDEV WAIT X: PRINT
Where, the value of x should be satisfied the following condition

$$\frac{INT\ X}{64} > \frac{20}{Baud\ rate} \quad (second)$$
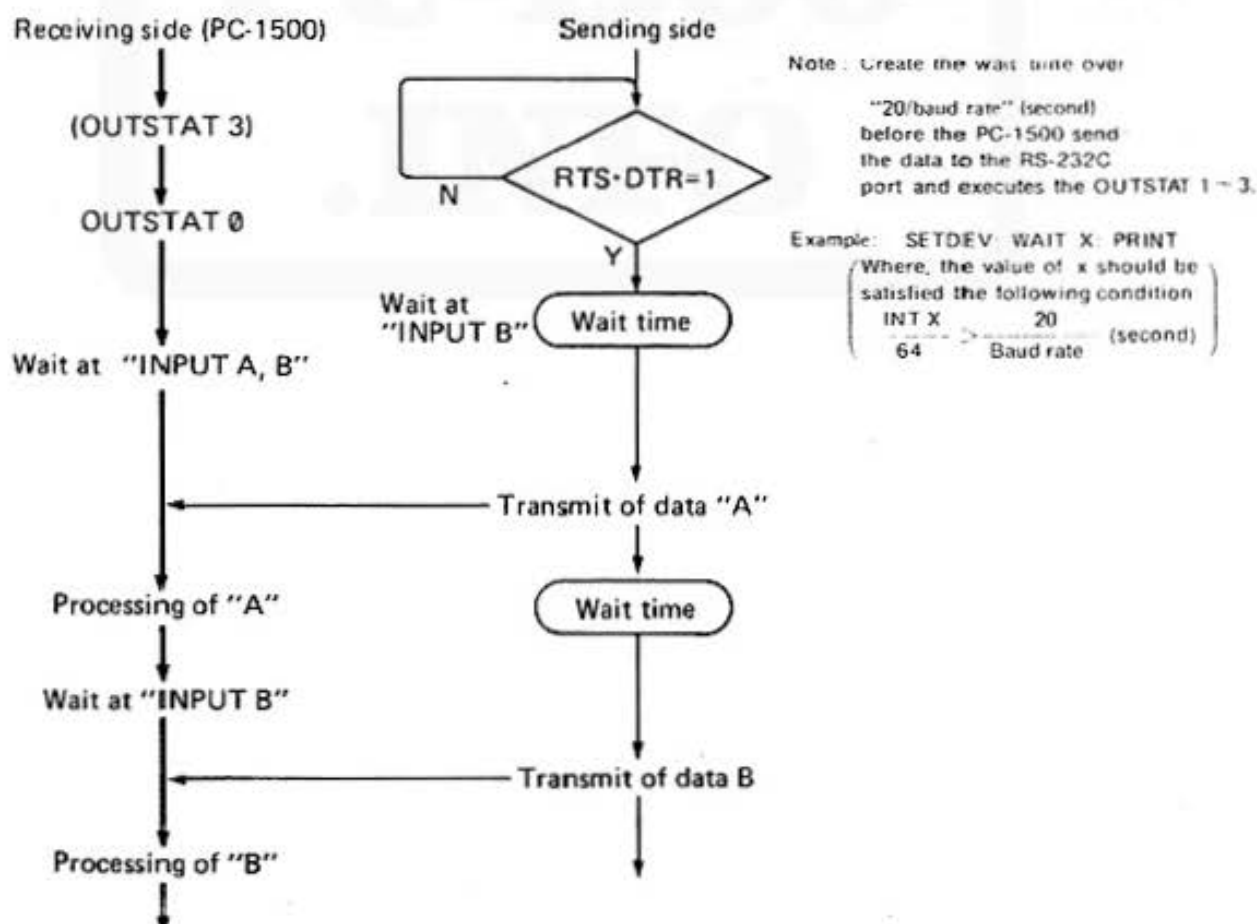
8. INPUT$

Almost similar to the INPUT statement described in the preceding page.
What differs is that the input data through the RS-232C port is not translated into the inter-
mediate computer language. Therefore, inclusion of "SIN 30" in the data before the CR code
appears will not be considered to be SIN of the trigonometric function, and that it does not
contain in the input data the function that involves arithmetical expression.

9. INPUT%

This command is also effective only when KI has been declared by the SETDEV statement.
● Data received through the RS-232C port are substituted with specified character array, of
which the contents are cleared upon execution of this command.
Command execution terminates when the CR code is received or when data are received
to the full capacity of character array.

[General format]
INPUT% character array name $(*)

(EX)   10   SETDEV KI

20   SETCOM 1200,7,E,1

30   OUTSTAT 0

40   CLEAR: DIM A$ (2,3)*1

50   INPUT% A$ (*)

60 END

Input data to the RS-232C port = RS-232C   CR

Contents of character array

A$ (I,J)	0	1	2	3
0	R	S	—	2
1	3	2	C	NUL
2	NUL	NUL	NUL	NUL

22

10. LPRINT

Executed only when PO has been declared by the SETDEV statement. Otherwise, it will execute the other option or cause ERROR 27.
- The specified data is sent out in the ASCII code.

(1) [General format]

LPRINT ⟶ expression ──────⟶

Sends out the END code after sending the contents of the expression in the format given by the USING statement.

(2) [General format]

LPRINT ──⟶ expression ──⟶ : ──────⟶
          └── ; ◄──┘   └──┘

When a semicolon (;) is used for a separator between expressions, data are sent out in succession.
When it ends with a semicolon (;), it will issue no END code.

(3) [General format]

LPRINT ──⟶ expression ──⟶ . ──────⟶
          └── , ◄──┘   └──┘

When data are separated by a comma (,), a blank portion to the next zone will be padded with SPACE codes.
If data were to be too long to contain in the same zone, it extends over a next zone.
Zones are defined in the manner shown below.
Capacity of each zone can be defined by means of the ZONE statement. (Default after power-on is 13 digits.)

END code output

(EX)   LPRINT "A","B","C"

NOTE-1:   (2) and (3) can be used in combination.

(EX)   LRPINT 10 ; 20 , 30

(4) [General format]

LPRINT TAB ⟶ expression ──⟶ :
                           └──⟶ End of statement

In regard with TAB, the next data will be outputted after issuing spaces to the position specified by the expression.

```
|←— Value of TAB expression ————→|——→Next data output
|                                |     |
|                                |—————|
|                                | spaces
|
END code output               TAB position
                              (Position next to the last output data)
```

If the TAB position is larger than the value of the 'TAB expression, the next data will be outputted from the position of TAB expression, after the output of the END code.

```
                              |←— Position of TAB expression —→|—→ Next data output
|                             |                                |   |
|                             |                                |———|
|                             |END code output
|
END code output              └(TAB position)
```

NOTE: The value of TAB expression must be smaller than digits/line specified by the CONSOLE statement.
It must be less than 255, if CONSOLE 0.

(5) LPRINT USING format designation;
USING can be used in the LPRINT statement.

(6) LPRINT
Only the END code is sent out.
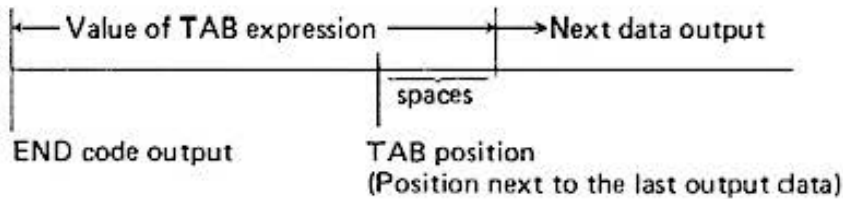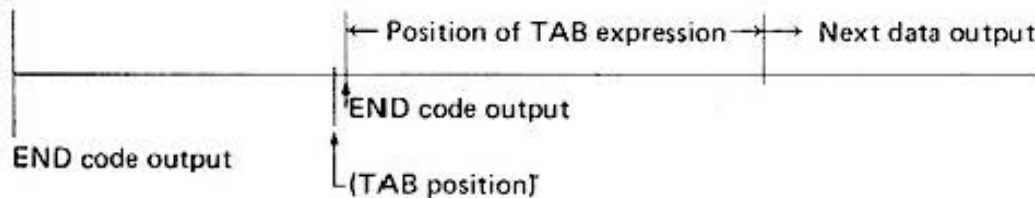
(EX)

Let's take up, for example, the video display terminal that incorporates the RS-232C interface.
As a single line of the display screen consists of 80 columns, automatic carriage return is executed upon receiving 80 characters of data.
It also causes the return of the carriage with the CR code.

(Program)

```
10  SETCOM  2400, 7, E, 1
20  SETDEV  PO
30  CONSOLE  0, 0
40  ZONE  16
50  LPRINT   "0"
60  LPRINT   "1" : "2" :
70  LPRINT
80  LPRINT   USING "###. #" : 1.234.
    USING ; -5. "ABCDEFGHIJKLMNOPQRS" , "Z"
90  LPRINT   "0". "1". "2". "3". "4". "5". "6". "7". "8".
    "9". "A". "B". "C". "D". "E". "F"
100 LPRINT   TAB(10) : " RS-232C "
110 END
```

Execution starts with "RUN ENTER ".

24

— Explanation —

10 line: Sets communication parameters.

20 line: Executes LPRINT to the RS-232C port.

30 line: The END code is not sent out on account of the size of the transmission data. The CR code is issued for the END code.
(Refer to the CONSOLE statement description.)

40 line: Size of an individual block in the LPRINT statement separated by a comma is specified to be "16".
(Refer to ZONE statement description.)

50 line: "0" is sent out, then the CR code.

60 line: "1" and "2" are sent out.

70 line: The END code is sent out.

80 line: Data are sent out in accordance with the USING statement definition.

90 line: Codes, "0" thru "F", are sent out in a unit of block.
The reason why ends of lines are on the same position is that 80 CRT screen columns could be divided without fraction by the parameter "16" defined by the ZONE statement.

100 line: Character string "RS-232C" is sent out after 11th position from the left.

NOTE: To send out the NULL code (= 00 hex), perform; LPRINT CHR$ (0).
Note that the following statement is ignored; LPRINT A$ (A$ = CHR$ (0)).

0 3 6 9 12 15 18 21 24 27 30 33 36 39 42 45 48 51 54 57 60 63 66 69 72 75 78

ABCDEFGHI JKLMNOPQRS Z

1.2

-5

1

6
B

RS-232C

0
12

2
7
C

3
8
D

2
8
D

4
9
E

11. LLIST

Effective only when PO has been declared by the SETDEV statement.
- Program is sent on the RS-232C port in the ACSII code.
- As the line number consists of five digits, space code(s) are prefixed to the line number when it is less than five digits. And, a space code is suffixed to the line number before sending out the text contents.

(EX)   10 REM ✱✱✱RS-232C ✱✱✱

65279 END

space	space	space	1	0	space	R	E	M	space	✱
✱	✱	R	S	–	2	3	2	C	✱	✱
✱	CR⁺⁾	6	5	2	7	9	space	E	N	D
space	CR⁺⁾									

"+)" represents the use of the END code in place of the CR code.

(1) [General format]
    LLIST
    The entire program is sent out.

(2) [General format]
    LLIST expression
    The program line specified by either the line number itself or by the label is sent out.

(3) [General format]
    LLIST expression-1, expression-2
    Program lines existing from the expression-1 to the expression-2 are sent out.
(4) LLIST, expression
    Program existing from the beginning of the program to the line represented by the expression is sent out.

(5) LLIST expression
    Program existing from the line represented by the expression to the end of the program is sent out.

NOTE: The sign ">" will be put on the display when there is no existing program.

(EX)

```
10    "A"REM  ✱✱✱RS-232C✱✱✱
20    OUTSTAT  0
30    CLS
40    WAIT  0
50    A$=RINKEY$
60    IF  A$  PRINT  A$ :
70    GOTO  50
80    END
```

```
10    "B"SETCOM  300,7,E,1
20    SETDEV  DO:OUTSTAT  0
30    IF  INKEY$  GOTO  30
40    B$=INKEY$
50    IF  B$  PRINT  B$ : :GOTO  30
60    GOTO  40
```

This program example contains two programs that are loaded using the MERGE statement.
With the first program, input of data through the RS-232C port is put on the display. With the second program, it sends out the respective code on the RS-232C port when there is a key entry.

Key operation	What is sent on the RS-232C port
SETDEV PO [ENTER]	Direct to execute the LLIST statement to the RS-232C port.
OUTSTAT 0 [ENTER]	It indicates after turning the RS-232C ports DTR and RTS signals active that the RS-232C interface is ready for operation.
LLIST [ENTER]	All programs lines.
LLIST 30 [ENTER]	30 CLS line only.
LLIST "B",30 [ENTER]	10 "B" SETCOM 300,7,E,1 20 SETDEV DO:OUTSTAT 0 30 IF INKEY$ GOTO 30
LLIST, 20 [ENTER]	10 "A" REM ***RS-232C*** 20 OUTSTAT 0
LLIST "B", [ENTER]	10 "B" SETCOM 300,7,E,1 20 SETDEV DO:OUTSTAT 0 30 IF INKEY$ GOTO 30 40 B$=INKEY$ 50 IF B$ PRINT B$;: GOTO 30 60 GOTO 40

NOTE: " √ " and "π" quoted in the program will be outputted by the code string of square plus a space and pi plus a space, respectively. The same is applicable for the CSAVEa statement which will be discussed later.
But, it will result in error if the CLOADa statement (which will be discussed later) is executed at the receive side with " √ " or "π" quoted in quotation marks.

12. PRINT
Effective only when DO has been declared by the SETDEV statement.
● Specified data is sent out in the ASCII code.

What differs from the LPRINT statement are as follows:
(1) Does not meet with the USING format (format free of USING format designation), so it will result in error if USING is inserted in this statement.
(2) TAB designation is not permitted.
(3) Though either a semicolon (;) or a comma (,) can be used for a data separator, the comma (,) is handled the same as the semicolon (;).
It is not possible to end with a comma (,).
(4) Sign code (+ or space) will not be affixed to the data of a positive value.

13. CSAVE, CSAVEa, CSAVEr, PRINT # transmission format

Except for CSAVEa, the contents of the memory is directly sent out with the header prefixed. Contents of the header may vary slightly as shown in the following figure.

With the CSAVEa, all internal codes are converted into the ASCII code before sent out, without need of the header in this case. Designation of the file name will be ignored as well.

A maximum size of a file name is 16 characters.

If the file name is shorter than 16 digits, blank portions will be padded with NULL codes (00 hex). Therefore, if no file name was designated, all 16 characters will be filled up with NULL codes.

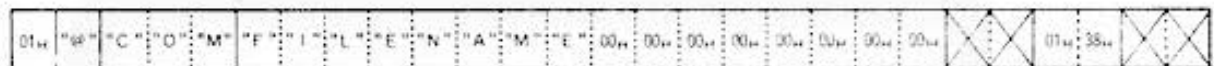Word length of communication parameter must be set to 8 bits.

Example: Example of the header

CSAVE (RUN, PRO mode) .... without file name designation (with program size of &0139)
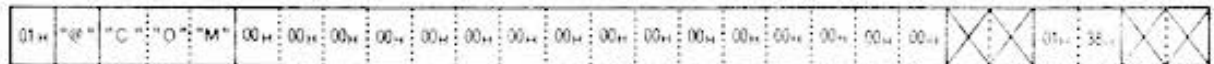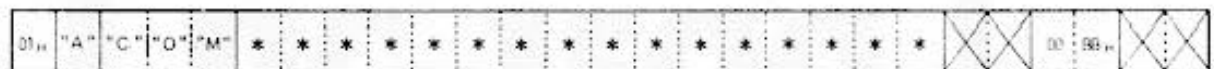

Capacity
−1

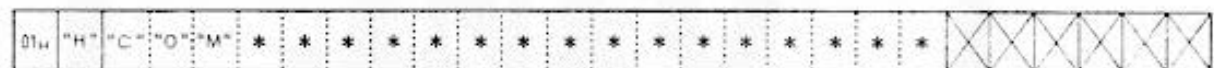CSAVE "FILENAME" (RUN, PRO mode) .... with file name designation.

Capacity
−1

CSAVE (RESERVE mode), CSAVEr

Capacity
−1

PRINT#

Note-1: Portions indicated with an asterisk (∗) will be filled up with the file name when there is a file name designated, but unoccupied portions will be padded with NULL code when no file name is designated.

Note-2: ⊠ is "don't care."

14. Loading by CLOAD, CLOADa, CLOADr, MERGE, MERGEa

Load statements, except CLOADa and MERGEa, will not allow proper loading (ERROR 61), unless the input header is correct. It would not start to load unless the input header matches with the specified file name.

With the CLOADa or MERGEa statement, it starts to load program at once, as it does not require intervention of the header.

IMPORTANT:

About 100 ms of all mark signal portion (no signal) will be required for an intergap between the header and the contents.
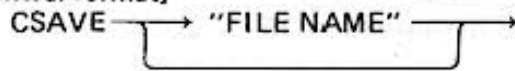
15. CSAVE

Effective only when CO has been declared by the SETDEV statement.
- Program in the RUN or PRO mode or reserve program in the RESERVE mode is sent out on the RS-232C port in the internal code (not the ASCII code).
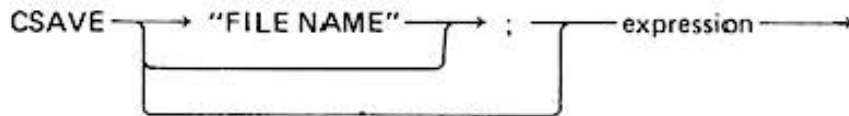
[RUN, PRO mode]
(1) [General format]

CSAVE ──→ "FILE NAME" ──→

A whole program is sent out.
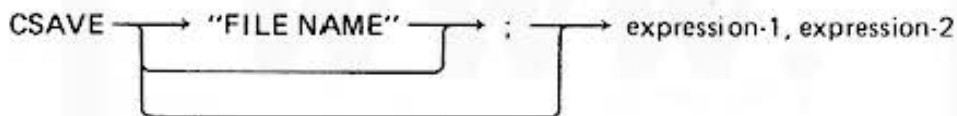
(2) [General format]

CSAVE ──→ "FILE NAME" ──→ ; ──→ expression ──→

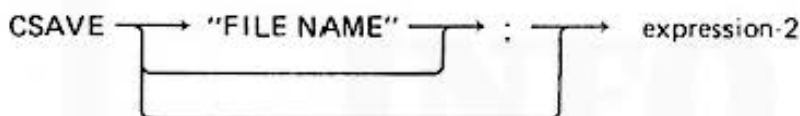Only the program line represented by the expression is sent out.
In case the contents of the expression is a character string (label), affix a semicolon to discriminate it from the file name.

(3) [General format]

CSAVE ──→ "FILE NAME" ──→ ; ──→ expression-1, expression-2

Program lines from the expression-1 to expression-2 are sent out.

(4) [General format]

CSAVE ──→ "FILE NAME" ──→ ; ──→ expression-2

Program lines from the beginning of the program to the expression-2 are sent out.

(5) [General format]

CSAVE ──→ "FILE NAME" ──→ ; ──→ expression, ──→
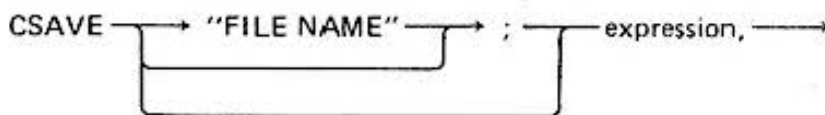
Program portion from the program line represented by the expression to the end of the program line are sent out.

NOTE-1: When there is no program to output with, the symbol " > " will be put on the display.

NOTE-2: If the given program line number was not found, the program line number closest to the line number, but larger, is assumed.

30

[RESERVE mode]
(6) [General format]

CSAVE ⟶⟶ "FILE NAME" ⟶⟶

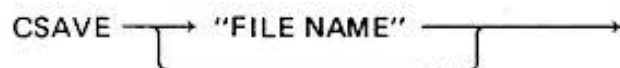A whole reserve program is sent out.

NOTE-3: Before execution of this statement, it needs to set the word length to 8 bits by the SETCOM statement.

16. CSAVEa

Effective only when CO has been declared by the SETDEV statement.
● Program is sent out in the ASCII code.

Though it uses the same general format as in (1) to (5) of the CSAVE statement, it would not send out subsequent portion of the merged program, if program in the memory were merged together. One more CR mode will be issued after sending the program.

Key operation	Output on RS-232C port
CSAVEa [ENTER]	10 "A" REM***RS-232C*** 20 OUTSTAT 0 30 CLS 40 WAIT 0 50 A$=RINKEY$ 60 IF A$ PRINT A$: 70 GOTO 50 80 END
CSAVEa:"B". [ENTER]	10 "B" SETCOM 300.7.E.1 20 SETDEV DO:OUTSTAT 0 30 IF INKEY$ GOTO 30 40 B$=INKEY$ 50 IF B$ PRINT B$::GOTO 30 60 GOTO 40

31

17. CSAVEr

Effective only when CO has been declared by the SETDEV statement.
- The contents of the reserve program is sent out on the RS-232C port in the internal code (same as the CSAVE statement in the RESERVE mode).

[General format]

CSAVEr ───┬─→ "FILE NAME" ──┬──────→
          └──────────────────┘

NOTE: Set the word length to 8 bits.

18. CLOAD

Effective only when CI has been declared by the SETDEV statement. (Only manual operation is permitted.)
- Program executed in the RUN or PRO mode or reserve program executed in the RESERVE mode is received through the RS-232C port in the internal code mode.

[RUN, PRO mode]
(1) [General format]

CLOAD ───┬─→ "FILE NAME" ──┬──────→
         └──────────────────┘

[RESERVE mode]
(2) [General format]

CLOAD ───┬─→ "FILE NAME" ──┬──────→
         └──────────────────┘

NOTE-1: Set the word length to 8 bits.
NOTE-2: Program will be NEW when an error or a break is met during loading operation.

19. CLOADa

Effective only when CI has been declared by the SETDEV statement. (Only manual operation is permitted.)
- Program is received through the RS-232C port in the ASCII code.

[General format]

CLOADa ──────→

- Length of one program line received through the RS-232C port must be 160 ASCII codes at a maximum and that it should be within 80 codes when converted into the internal code.
- When the program is to be loaded from the RS-232C port, secure two seconds of all MARKs (no signal) as an intergap between program lines).
- When the line is headed by the CR code, it will terminate program loading operation.

NOTE: When an error or a break is met during loading of the program, the program portion already loaded in the memory stays stored in the memory.

32

20. CLOADr

Effective only when CI has been declared by the SETDEV statement. (Only manual operation is permitted.)

● Reserve program is loaded from the RS-232C port in the internal code.
Same function as the CLOAD statement in the RESERVE mode. The word length must be set to 8 bits.

21. MERGE

Effective only when CI has been declared by the SETDEV statement. (Only manual operation is permitted in the RUN or PRO mode.)

● Input program (intermediate program code) from the RS-232C port is merged with the program currently existing in the memory.

[General format]

    MERGE ───┬──→ "FILE NAME" ──┬──→
             └───────────────────┘

NOTE-1: Set the word length to 8 bits.
NOTE-2: Only the loaded portion will be cleared, when an error or a break is met during loading operation.

22. MERGEa

Effective only when CI has been declared by the SETDEV statement. (Only manual operation is permitted in the RUN or PRO mode.)

● ASCII code program sent from the RS-232C port is linked with the program currently existing in the memory.
Rest of other details are the same as CLOADa.

[General format]

    MERGEa ──┬──→ "FILE NAME" ──┬──→
             └───────────────────┘

NOTES:
1. Only the loaded portion is linked when an error or a break is met during loading operation.
2. There will be no practical problem even if an insignificant program is put on the display by depression of ↑ or ↓ key (PRO mode),
   ● after interruption of command execution by means of ON key, or,
   ● after occurrence of an error during command execution,
   immediately after starting execution of MERGEa (before the transfer of program). (The contents of such a program will never be executed.)
   This condition occurs when ↑ key is pushed first when the program is not on the display. In such an event, either push ↓ key after the depression of CL key or call the desired line by means of the LIST statement. It is not possible to correct or insert program under this condition. Else, the modified or inserted program will be linked after the existing program.

## 23. Data input and output by PRINT# or INPUT# statement

(1) Set the word length to 8 bits.
(2) Header will be sent out prior to the transfer of the contents when the PRINT# statement is executed.
   Except when the file name has not been specified, headers must be identical upon execution of the INPUT# statement.
(3) It will require an intergap (MARK state) of about 4 seconds long between the header and the variable and between variables in executing the INPUT# statement.

24. PRINT#

Effective only when CO has been declared by the SETDEV statement.
● The contents of the specified variable is sent out on the RS-232C port in the internal code.

[General format]

PRINT# ⟶ "FILE NAME" ⟶ ; ⟶ variable name ⟶
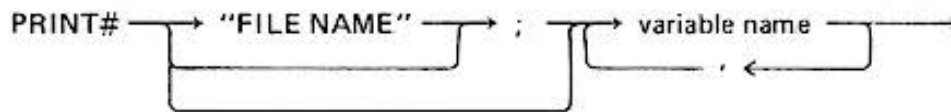
For example, variable name for a variable array should be "A1$(*)", which will cause the sending of the entire variable array of A1$. Note that it is not possible to send the contents of the variable array one by one; it will result in ERROR 1.
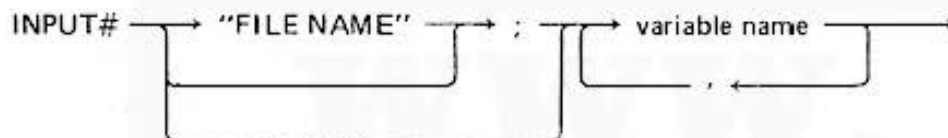To send all fixed variables, write "@(*)"for (A − Z) or "@$(*) " for (A$ − Z$).

25. INPUT#

Effective only when CI has been declared in the SETDEV statement.
● The data received through the RS-232C port is substituted with the specified variable in the internal code mode.

[General format]

INPUT# ⟶ "FILE NAME" ⟶ ; ⟶ variable name ⟶

If the variable name is a variable array, it should have been declared by the DIM statement.
Two-letter variable name will be defined automatically.
The variable name can be specified in the same manner as in the PRINT# statement.

26. INPUT#-8,

This command is executed without any connection with KI that was declared by the SETDEV statement.
It has the same execution as in INPUT, INPUT$, and INPUT% statement for which KI has been declared by the SETDEV statement.

    INPUT#−8,   =   SETDEV KI
                       INPUT

    INPUT#−8, $ =   SETDEV KI
                       INPUT$

    INPUT#−8, % =   SETDEV KI
                       INPUT%

27. PRINT#-8,

This command is executed without any connection with PO that was declared by the SETDEV statement.
It has the same execution as in the LPRINT statement for which PO has been declared by the SETDEV statement.

    PRINT#−8, =   SETDEV PO
                       LPRINT

28. TRANSMIT

LONG SPACE (Space Signal) will be sent on the RS-232C port, as many as the specified value (Padding).

[General format]

    TRANSMIT BREAK, expression ⟶

Where, expression = 1 ~ 255.

The time required for the transmission of LONG SPACE nearly equals INT expression/64 seconds.

29. RINKEY$

It is the nonnumeric function that furnishes the last one byte of information that existed on the RS-232C port immediately before the execution of this command.
A NULL code (00 hex) will be issued when there is no input code.

30. ZONE

This command is used to determine the block length of the output data that is sent on the RS-232C port, when data are separated by a comma (,) in the LPRINT statement.
Default after power on is the block length of "13".

[General format]

    ZONE ⟶ expression ⟶

Where, expression = 1 ~ 31.

31. ERN

It is the function with which the error code is provided within a range of 0 to 255.

32. ERL

It is the function with which the error line number is provided.
0 :   No error after program execution.
1 ~ 65279: Error line number.

33. SPACE$

A string of spaces composed of a given number (0 ~ 32).

[General format]

    SPACE$ ⟶ numeric expression ⟶

34. FEED

Used to send the END code, which will be a CR code or a LF code or combination of both, which should be specified by the CONSOLE statement.

(1) [General format]

        FEED ⟶

Only one END code is sent.

35

(2) [General format]

FEED $\longrightarrow$ expression $\longrightarrow$

Where, expression = 1 ~ 65535.

END codes are sent out, as many as specified (integer part).

NOTE: If the transmission should end with the END code before sending out the END
code by this command, the END code by this statement will be replaced with a
SPACE code plus an END code.

35. CONSOLE

1. With this command, the number of digits per line is specified, that is, the END code will be
issued upon sending out the data of the specified size.
2. Used to specify the kind of END code in sending out the data on the RS-232C port by the
LPRINT, PRINT, LLIST or FEED statement.
Default, after power on, is the CR code.

(1) [General format]

CONSOLE $\longrightarrow$ expression $\longrightarrow$

Where,

$$\text{Expression} = \begin{cases} 0: & \text{Number of digits per line is unlimited.} \\ 16 \sim 255: & \text{Number of digits per line.} \end{cases}$$

(2) [General format]

CONSOLE expression-1 , expression-2 $\longrightarrow$

The expression-1 is the same as in (1).
The kind of END code is specified with the expression-2.

$$\text{Expression-2} = \begin{cases} 0: & \text{CR code for the END code.} \\ 1: & \text{LF code for the END code.} \end{cases}$$

(3) [General format]

CONSOLE expression-1 , expression-2 , expression-3 $\longrightarrow$

The expression-1 is the same as in (1).
The kind of END code is determined with a combination of the expression-2 and expression-3.

Expression-2	0	0	1	1
Expression-3	0	1	0	1
END code	CR+CR	CR+LF	LF+CR	LF+LF

36

TERMINAL PROGRAM MODE

When the TERMINAL statement or DTE statement is executed, the BASIC program mode changes to the terminal program mode.

The following functions will be executed in the terminal program mode.
(1) Output of keyboard data on the RS-232C port.
(2) Indication of that data on the display.
(3) Output of the RS-232C port input data on the display (or through the parallel port).
(4) Confirmation of that input data in a unit of word or line.
(5) Output of a specific part of RS-232C input data on the CE-150 printer (or through the parallel port).

And, choice of following function is possible in the menu select mode.
(1) Choice of self ECHO (whether the keyboard data be put on the display or not).
(2) Choice of automatic sending of the XON/XOFF code.
(3) Auto sign start.
(4) Defining auto sign code.
(5) Defining five soft keys.
(6) Defining communication parameters such as baud rate, word length, parity assignment, stop bit length.
(7) Choice of normal or auto paged (halt of communication at every 512 bytes) or auto line (halt of communication at every line).
(8) Specifying the receive data output channel, CE-150 or parallel port.
(9) Determing whether the received data is to be put on the display or sent out through the parallel port.
(10) Sending of ETX and LONG SPACE.

NOTE: If the data transfer speed was set faster than 600 bauds in the TERMINAL PROGRAM mode, it may be too fast to follow, sometimes. But, it can be operated above 600 bauds if it was set in the XON/XOFF mode, which the transmission is stopped by the reception of the XOFF code (CONTROL S) and the transmission is resumed by the reception of the XON code (CONTROL Q).

1. TERMINAL

Command used to go into the terminal program mode.

[General format]

TERMINAL ⟶

2. DTE

Command used to go into the terminal program mode.

[General format]

DTE ⟶

- In order to operate under the terminal program mode, it requires a free area* of more than 570 bytes that are not used for the BASIC program and variables, out of 1850 bytes user's area for the basic memory size, 5946 bytes area for the memory expanded with the 4 K option RAM, or 10042 bytes area for the memory expanded with the 8 K option RAM. It will cause ERROR 51 if it is less than 569 bytes.

  - : The value of free area is obtained by "STATUS 3 – STATUS 2".
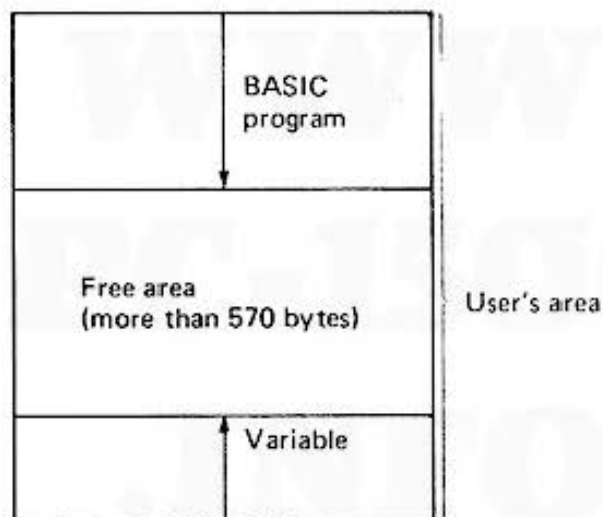
When ERROR 51 is caused, it will be not possible to go into the terminal program mode, unless the error is removed by the CLEAR or NEW statement.

3. Difference between TERMINAL statement and DTE statement

Basically, these two statements are identical, but there are some differences in their initial states as described below.

		TERMINAL	DTE
Communication parameters	Baud rate	Value at entry	300
	Word length	↑	7
	Parity	↑	E
	Stop bit	↑	1
Operation		Normal operation	Normal operation
XON/XOFF		ON	OFF
ECHO		OFF	ON
Print		CE-150	CE-150
TRACE function		OFF	OFF
Display		Clean text (Word Wrap)	Clean text (Word Wrap)
CL , SHIFT CA/CL		Non-operation	EXT code, LONG SPACE

4. Symbols

Symbols "RUN" and "III" will be active during the time of the terminal mode.
Eleven symbols of the computer may have special significance in the terminal program mode.

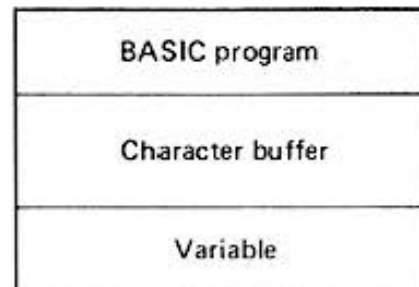| Symbol	State	Function
BUSY	ON	The computer is not ready to receive.
	OFF	The computer is ready to receive.
SML	ON	When any key A thru Z is pushed while this symbol is on, the small letter code corresponding to the key depression (a ~ z) is entered. When the SHIFT symbol is on, it enters the corresponding capital letter (A ~ Z).
SHIFT	ON	Indicates that the [shift] key function is on. (Alternate action)
DEG	ON	Indicates the state of the special key ' (MODE III) activation with which ASCII code not available on the keyboard is generated when this symbol is on. (Alternate action)
RAD	ON	Control code is generated when this symbol is on. (Alternate action)
RUN PRO RESERVE	ON ON ON	Indicates the normal operation. Indicates the auto paged operation. Indicates the auto line operation.
I	ON	Indicates MODE I of software key.
II	ON	Indicates that software key is ready to assign TRACE, NORMAL, AUTO PAGED, or AUTO LINE operation.
III	ON	Indicates that software key is ready to be used as special defined key.

5. Code chart

L\U	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			SPACE	0	@	P	`	p								
1			!	1	A	Q	a	q								
2			"	2	B	R	b	r								
3			#	3	C	S	c	s								
4			$	4	D	T	d	t								
5			%	5	E	U	e	u								
6			&	6	F	V	f	v								
7	BEL		'	7	G	W	g	w								
8	BKS		(	8	H	X	h	x								
9			)	9	I	Y	i	y								
A			*	:	J	Z	j	z								
B			+	;	K	[	k	{								
C			,	<	L	\	l	!								
D	CR		−	=	M	]	m	}								
E			.	>	N	↑	n	~								
F			/	?	O	←	o	■								

NOTE: Code not assigned in 00 thru 1 F (hex) has no function except for BEL, BKS and CR. Codes 80 thru FF (hex) are used for graphic pattern application (see Graphic functions).

6. Character buffer

Received data and key input data during ECHO ON is stored in the memory of the computer (which is a part of the user's area that is not used for the program area and variable area) called a character buffer. Size of this character buffer must be more than 570 bytes. It will result in ERROR 51 unless this size area were not secured.

```
┌─────────────────────────────┐
│                             │
│      BASIC program          │
│                             │
├─────────────────────────────┤
│                             │
│      Character buffer       │
│                             │
├─────────────────────────────┤
│                             │
│      Variable               │
│                             │
└─────────────────────────────┘
```

In such an event, it needs to clear variables using the CLEAR statement or to clear the BASIC program and variables using the NEW statement. If it is undersierable to clear, save such variables and program on the cassette tape via the CE-150.
The following maximum capacity will be available for the character buffer when the NEW statement has been executed.

1792
5888 (4 K RAM module CE—151, in use)
9984 (8 K RAM module CE—155 in use)

Data are stored in the character buffer from the youngest memory address, and incoming data chases out the oldest data when the character buffer becomes fully occupied, so that the latest data will remain in the character buffer at any time.

Character buffer

```
        ┌───────────────────────────────────┐
     ◄──┤                                   ├──◄──  Newest data
        └───────────────────────────────────┘
         └── Oldest data
```
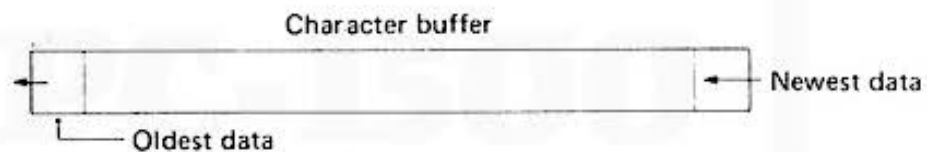
7. Keys

Except in the menu select mode, any code listed in the code chart can be generated in the terminal program mode. However, codes, 80 – FF, cannot be generated, except the time when defined to the software keys, F1 – F5, of MODE I. (Refer to Menu select mode setup.)
Codes not inscribed on keytops can be generated in a following manner. Use the supplied template to identify the definition assigned to the keys.

Code	Name			Key operation (MODE III)
00	NULL			[•]
01	CONTROL	A	(SOH)	[A]
02	"	B	(STX)	[B]
03	"	C	(ETX)	[C] or [CL]
04	"	D	(EOT)	[D]
05	"	E	(ENG)	[E]
06	"	F	(ACK)	[F]
07	"	G	(BEL)	[G]
08	"	H	(BKS)	[H] or [ ]
09	"	I	(HTAB)	[I]
0A	"	J	(LF)	[J]
0B	"	K	(VTAB)	[K]
0C	"	L	(FF)	[L]
0D	"	M	(CR)	[M] or [ENTER]
0E	"	N	(SOUT)	[N]
0F	"	O	(SIN)	[O]
10	"	P	(DLE)	[P]
11	CONTROL	Q	(DCI)	[Q]
12	"	R	(DC2)	[R]
13	"	S	(DC3)	[S]
14	"	T	(DC4)	[T]
15	"	U	(NAK)	[U]
16	"	V	(SYN)	[V]
17	"	W	(ETB)	[W]
18	"	X	(CAN)	[X]
19	"	Y	(EM)	[Y]
1A	"	Z	(SUB)	[Z]
1B	ESCAPE			[ ]
1C	FS			[Q]
1D	GS			[W]
1E	RS			[E]
1F	VS			[R]
27	'			[T]
5B	[			[U]
5C	\			[L]
5D	]			[I]
5E	↑			[O]
5F	←			[P]

Code	Name	Key operation (Mode III)
6 0	`	K
7 B	{	H
7 C	¦	F
7 D	}	J
7 E	~	G
7 F	■	D

On the contrary, codes inscribed on keytops " ∧ ", " ⌐ ", and " π " can not be generated.
[SHIFT] [↑] and [SHIFT] [↓] work as window key. (Refer to Window key description.)
[SHIFT] [SPACE] have no function.

NOTE: In the MENU SELECT mode, only those inscribed on the side of the keytop can be put on the display.
Keyboard layout used the supplied template is shown in a page to follow.

8. Repeat function

Almost all keys are provided with the repeat function in the terminal program mode (after entry of [↑] or [↓] key.
When a repetitive key is held depressed continuously, it starts to repeat the same key entry in about one second later.

(EX)

Key operation	Display
[A]	A _
One second after [A]	A A A A A _
[SHIFT] [A]	A A A A A a _
One second after [A]	A A A A A a a a a _

The following keys have no repeat function.
● Software key of MODE II and MODE III.
● Window keys during data transfer (→ , ← , ↑ , ↓).
● CL , [SHIFT] CL keys when CL = ETX ON.
● [SML] and [SHIFT] keys.

9. Invalid keys

The following keys become invalid in the terminal program mode.
[DEF] , [OFF] , [MODE] , [SHIFT] [MODE] , [SHIFT] [SPACE] .
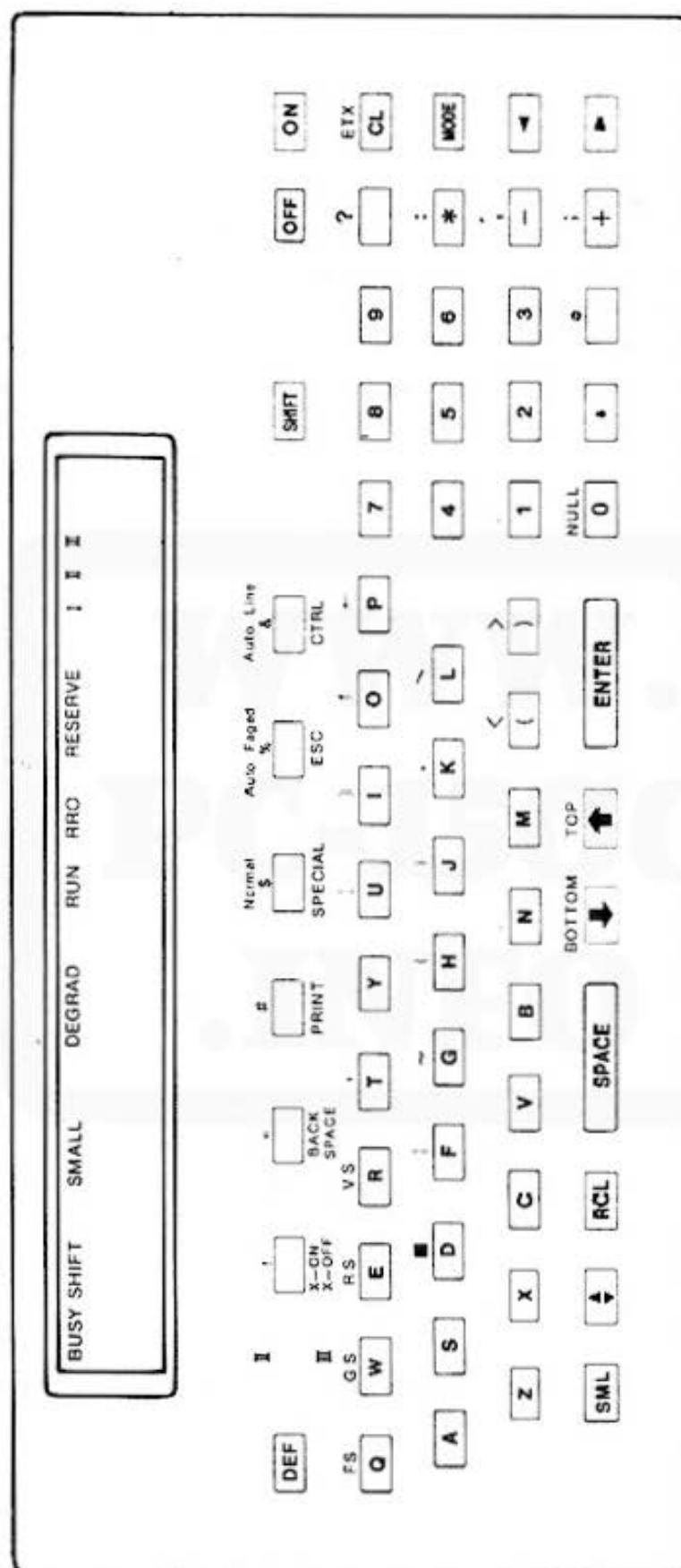
PC-1500 with the supplied template
(Effective only when the terminal mode is executed.)

10. Special keys

[1] Software keys

Three kinds of RESERVE modes, I, II, III, must be selected by means of [⬦] key.
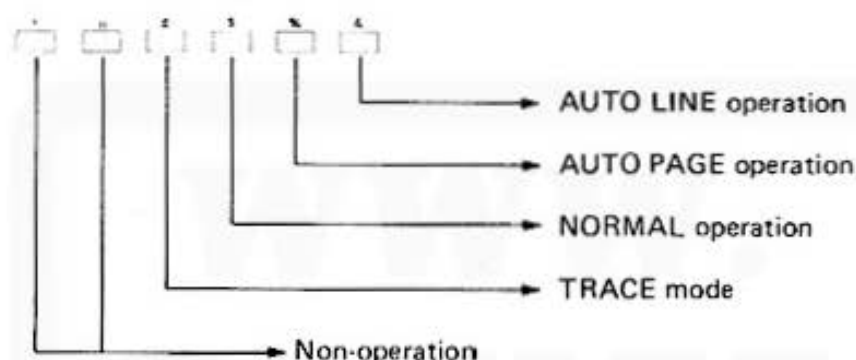
1) MODE I

User defined key sequence is executed in this mode, except for the F6 key.

Contents can be retained in the BASIC program mode during the terminal program entry. For change in the terminal program mode, refer to the Menu Selection Mode Setup description.

If command, statement and function such as RUN, GOSUB, SIN, etc., were substituted in the BASIC program mode, they will not be handled in a correct manner in the terminal program mode.

2) MODE II

Function such as TRACE mode, NORMAL operation, AUTO PAGED operation, and AUTO LINE operation can be selected during the terminal program mode. (See Menu Select description.)


3) MODE III

The following functions are defined in the terminal program mode.

⬚ : Resume and stop receiving operation (alternate action).

When this key is pushed while making communication with the host computer with the BUSY symbol off, it halts the receiving operation. If XON/XOFF were ON in this occasion, it will send out the XOFF code (CONTROLS). Then, the BUSY symbol is activated and the cursor disappears from the display. Note that it may take some time before the cursor goes out.

Depression of the key in this condition makes the BUSY symbol go out and the communication will be enabled with the host computer.

NOTE: Key entry in the ECHO OFF condition while reception is at halt will send out that key code. In the ECHO ON condition, key entries are accepted as many as 40 bytes maximum, and they will be sent out when transmission is enabled.

⬚ : Generates the BACKSPACE code (08 hex).

⬚ : PRINT START/STOP key (alternate action)

When this key is pushed while communication is halted with the BUSY symbol on, it sends out on the parallel port or on the CE—150 printer (EXT OFF) all received data in the character buffer, or the contents of key entry, if ECHO ON. If the key is pushed again during output, it halts the output.

Data will be sent out from the beginning of the character buffer after the stop of reception operation. After the search of the character buffer using the window key (see Window key description), succeeding data to the display contents will be outputted.

45

⌐¬ : Special key (alternate action)
Used to generate character code which is not inscribed on the keytop, except for control codes, 00-1AH.
To generate the " { " code, push ⌐¬ key, which will cause the DEG symbol to activate, then push ⌐H¬ key which will extinguish the DEG symbol. symbol.

⌐¬ : Used to generate ESCAPE code (1BH).

⌐¬ : Used to generate control code, 00-1AH.
To generate the ETX code (03H), push ⌐¬ key which will cause the RAD symbol to activate, then push ⌐c¬ key which will extinguish the RAD symbol.

> NOTE: Contents of the MODE II and MODE III has nothing to do with the reserve memory.

[2] ⌐↕¬ key

Used to change RESERVE mode, I, II, III.
It is effective during data transfer with the host computer or during output of character buffer contents on the CE-150 or parallel port.

[3] ⌐RCL¬ key and key symbols

Effective only during receive operation when the BUSY symbol is on.
When this key is pushed, key symbols are activated with respect to the symbol I, II, or III in activation.

MODE I: Contents that the user had defined in the BASIC program mode.
MODE II: Operational assignments are put on the display, indicating assignment of the following four keys in MODE II (effective even during the time of transmission).

			Nrm	A/P	A/L
⌐¬	⌐¬	⌐¬	⌐¬	⌐¬	⌐¬
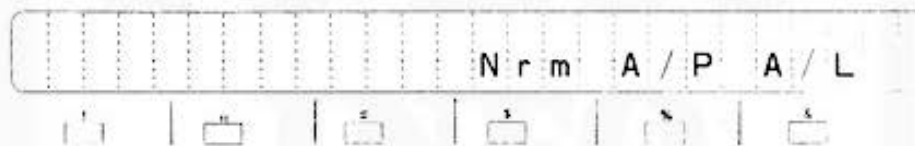
1) ⌐¬ : Trace function on/off key (alternate action). See Trace function description.

2) ⌐¬ : Normal operation (RUN symbol in activation).

3) ⌐¬ : Auto paged operation (receiving operation stops each time 512 bytes are stored in the character buffer).

4) ⌐¬ : Auto line operation (receiving operation stops each time the CR code is transferred to the character buffer).

MODE III: Puts on the display six software abbreviations that are defined in this mode.

X0/0	Bks	Prt	Spcl	Esc	Ctl
⌐¬	⌐¬	⌐¬	⌐¬	⌐¬	⌐¬
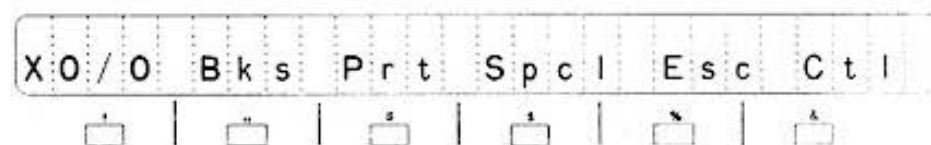
When the ⌐RCL¬ key is pushed again, it returns to the previous display.
When ⌐↕¬ key is pushed while a key symbol is on, it changes MODE (I, II, III), and key symbols pertinent to that mode are put on the display.

46

[4] Window key

Six window keys, [►] , [◄] , [▲] , [▼] , [SHIFT] [▲] , and [SHIFT] [▼] are available.
It will halt the receiving operation, when a window key is pushed during data transfer with the BUSY symbol out. In other words, it causes the same result as if   (MODE III) is pushed. However, it is not possible to halt the receiving operation by window key in AUTO LINE operation and TRACE modes.

1) When the operation is suspended, the top line of the character buffer is displayed.

2) When a window key is pushed after the receiving operation is stopped by   key (MODE III) with the last contents of the character buffer on the display, the top line of the character buffer is displayed.

3) When a window key is pushed in the condition of (1) or (2), the following action will take place.

    [►] :   The word next to the currently displayed contents will be put on the display.

    [◄] :   The word immediately preceding the currently displayed contents will be put on the display.

    [▲] :   The line immediately preceding the current line will be put on the display. If the line were to be too long, only the leading part will be displayed.

    [▼] :   The line next to the current displayed line will be put on the display. If the line were to be too long, only the leading part will be put on the display.

    [SHIFT] [▲] :   The top line of the character buffer will be put on the display.

    [SHIFT] [▼] :   The last part of the character buffer will be put on the display as many as possible.

(EX)

T	h	i	s		i	s		J	a	m	e	s		B	r	o	w	n	e		s	p	e
a	k	i	n	g	.	↵	I		w	a	n	t		t	o		t	a	l	k		w	i
t	h		T	o	m	.	↵	-	-	-	-	-	-	-	-	-	-	↵	P	l	e	a
s	e		l	e	a	u	e		m	y		m	e	s	s	a	g	e		t	o		h
i	m	.	↵	I		a	m		g	o	i	n	g		t	o		c	o	n	c	e	r
t	.	↵																					

Key operation	Display
	`going to concert.`
◄	BUSY `This is James Browne`
►	BUSY `is James Browne speaking.`
►	BUSY `James Browne speaking. I`
SHIFT ↓	BUSY `. I am going to concert.`
SHIFT ↑	BUSY `This is James Browne`
↓	BUSY `I want to talk with Tom`
↑	BUSY `This is James Browne`
↑	BUSY `This is James Browne`
↓	BUSY `I want to talk with Tom.`
◄	BUSY `speaking. I want to talk`
☐ (MODE III)	`. I am going to concert. _`

NOTE: The CR code will be displayed as a space.

[5] $\boxed{\text{CL}}$, $\boxed{\text{SHFT}}$ $\overset{\text{CA}}{\boxed{\text{CL}}}$

With CL = EXT ON in the menu mode, depression of $\boxed{\text{CL}}$ key will send out the FTX code (03H), and, depression of $\boxed{\text{SHFT}}$ $\overset{\text{CA}}{\boxed{\text{CL}}}$ keys will sent out LONG SPACE for a period of about 240 milliseconds (Padding).
With CL = ETX OFF, these two depressions will become inoperative.

[6] $\boxed{\text{ON}}$

With the depression of this key, the control returns to the MENU SELECT mode, regardless of current operating condition.

[7] Paper feed key

PAPER FEED key of the CE-150 is only effective when the BUSY symbol is on.
But, it becomes inoperative, when the CE-150 is in a low battery condition.

11. Transmission codes

(1) Space

Space is used for a word separator.
Space after the CR code will not be put on the display, but it will be sent out to the CE-150 or through the parallel port.

(2) CR

CR is used for a word and line separator.
CR is displayed in a same manner as in space.
When sent to the CE-150, it performs carriage return and linefeed.
When sent through the parallel port, either the CR or LF code is outputted, the selection defined by the first END code designation of the CONSOLE statement.
Default after power on is the LF code.
The CR code appearing after the first CR code will be ignored. Existence of codes, 00H ~ 1FH, instead of the CR code will also be assumed as continuing appearance of the CR code.

(3) BEL

With reception of the BEL code or with depression of $\boxed{}$ $\boxed{\text{a}}$ (MODE III) keys during ECHO ON, it sounds the bell for a period of about 150 milliseconds. However, the bell will be stopped when data is received within that period.
Bell is disabled when BEEP OFF was executed in the BASIC program mode.

(4) BKS

With reception of the BKS code or with depression of $\boxed{}$ (MODE III) key during ECHO ON, it causes the following actions:
● One digit or one graphic of data disappears from the display, and
● "$\backslash$" (=5CH) is sent through the parallel port at TRACE ON.
Though the BKS code deletes the newest data in the character buffer, the disappeared data will not be necessarily be identical with what is on the display.

(5) ESC

The ESC code (1BH) will only be effective when it is followed by "I".
When "ESC I" is received, it will send out the code string of "#PC2 CR".

(6) Other control codes

They are ineffective codes.

12. Display

During the time of CLEAN TEST OFF, receiving of data or key entry in the ECHO ON mode will cause horizontal scrolling.

```
This  is  James  Browne _
```

```
This  is  James  Browne .spea_
```

```
s  James  Browne  speaking . _
```

On the other hand, the following action will take place in the case of CLEAN TEXT ON. (WORD WRAP).

(1) The display contents are retained until a next space or CR code is received, if a space or CR code of the received data was found within 21st to 26th display column. With appearance of the next space or CR code, the present display contents are cleared and replaced with the newly received data.
   Unless a next space or CR code was not received within 26 digits, it puts the received data on the display.

(2) If a space or CR code was not received within 21st to 26th column, it performs horizontal scrolling.

```
This  is  James  Browne  _
```
Received data = speak

```
This  is  James  Browne  _
```
Received data = ing. CR

```
speaking .  _
```

(3) When a key entry is met during holding of the display contents until a space or CR code is received after receiving of the space or CR code on 21st to 26th column, it displays the suspended data after a horizontal scrolling.

```
This  is  James  Browne  _
```
Received data = speak

```
This  is  James  Browne  _
```
Key entry = [1]

50

```
h i s   i s   J a m e s   B r o w n e   s p e a k _
```
Received data = ing. CR

```
s   J a m e s   B r o w n e   s p e a k i n g .   _
```
Received data = I

```
s   J a m e s   B r o w n e   s p e a k i n g .   _
```
Received data = space

```
I   _
```
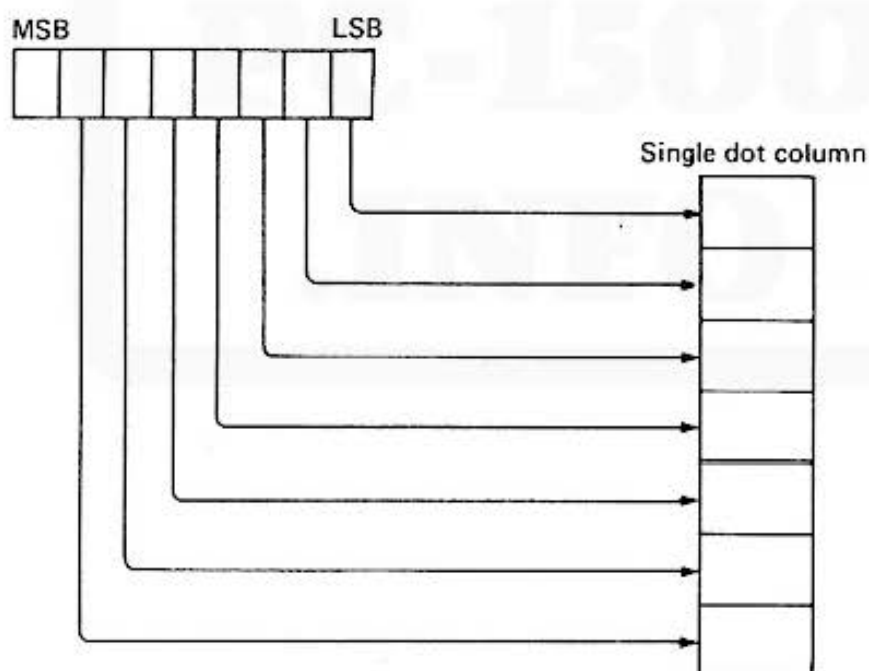
NOTE: Received data with error (Parity, Overrun, Frame) is displayed for?

13. Graphic codes

Codes 80H thru FFH are used for graphic application.
When any of these codes is received, it will display in the same format as described in the GPRINT statement.

A row of bits will be made active (bit = 1) or inactive (bit = 0) according to bit arrangement after the most significant bit (MSB) of the received data.
In this scope, the character length must be set to 8 for the graphic application.
Though a graphic code is not sent to the CE-150, it will be sent out through the parallel port.

51

14. Menu mode

It goes into the menu mode after one of following operations:
(1) When the TERMINAL or DTE statement is executed.
(2) When ⌐ON┐ key is pushed in the terminal program mode.

The following message is put on the display before going into the MENU SELECT mode.

```
___ENTER MENU SELECTION ..
```

15. Menu select mode

The warning message will be put on the display in the following conditions:
(1) After execution of Menu mode.
(2) When the POWER switch of the CE-158 was off in executing the terminal program or when there is a need of recharging the battery.

```
LOW BATTERY
```

Also, the following message will be put on the display in the following condition:

(3) If the parallel port BUSY signal was high for 10 seconds in sending out the buffer data onto the parallel port during execution of the terminal program.
The BUSY signal will be in high level when no connection is made with the parallel port, when power was not turned on to the connected device, or when the connected device is not working properly.

```
LPRINTER ERROR
```

To be followed by:

```
Terminal:    Ent Aut Quit
```

In the terminal selection mode a variety of conditions are set up such as the baud rate, echo selection, etc.

(1) Terminal

Depression of ⌐▪┐ , ⌐▪┐ , or ⌐▪┐ key when the following message is on the display will enable the following functions.

```
Terminal:    Ent Aut Quit
```

1) ⌐▪┐
Terminal program will be executed after activating DTR and RTS of the RS-232C port. The display is cleared at this point, with the BUSY symbol turned out.

2) ⌐▪┐
This code will be sent out, if the auto sign code was defined after the execution of (1).
When the CE-150 is in connection in executing above
(1) and (2), the followings will occur:
(a) Single linefeed.
(b) CSIZE set to "2" (18 characters/line).
(c) Left to right printing direction is set.

52

3)

Returens from the terminal program mode to the BASIC program mode.
The display at this point will be as follows:

```
          DEG   RUN                        I
>
```

(2) · Setup

Depression of [↓] key while the terminal message is on the display will put the setup message on the display.

```
Setup:              Aut  Fnc  Com
```

1)

The auto sign code is defined with this key.
The following prompt will be put on the display when this key is depressed.

```
CHANGE AUTO SIGN ? (Y/N) _
```

Push [↓], [ENTER], [N] [ENTER] when no change is required.
Then, the display returns to the setup request.
Push [Y] [ENTER] to make change, then enter the desired description (i.e. RS-232C).
(The display contents will shift from right to left.)

```
AUTO SIGN ? (Y/N) Y RS-232C
```

Then, push [ENTER] key.

```
CHANGE AUTO SIGN
```

It will return to the setup display after above message.
When [↓] key is depressed during entry of the auto sign code, it will put "@" on the display. Same effect when [SHIFT] [=] are pushed. "@" will become the CR code at the time of transmission.
As the auto sign code is secured in the reserve area, it will display the error message described below, when it is not possible to hold in the reserve area. Note that it is not possible to secure in the reserve area even in the BASIC porgram mode.

```
SETTING ERROR
```

(NOTE) The auto sign code is cleared when the reserve program is programmed newly or is loaded in the BASIC mode.

2)

Used to define software keys in MODE I.
The MODE I contents that are used for the BASIC mode can also be used in the terminal mode, but, its contents can be changed or defined in this stage.
Commands, statements, and functions such as RUN, GOSUB, SIN that are defined in the BASIC program mode are outputted in the internal code mode, not the ASCII code mode. This impedes correct data output.
If there was nothing defined to "F1", depression of [ ] key will bring the following message on the display.

```
C H A N G E   F 1 =   ?  ( Y / N )  _
```

Depression of ⟨ N ⟩ ⟨ENTER⟩ returns the display to the setup request. Depression of ⟨↓⟩ or ⟨ENTER⟩ will make it advance to "F2".
To define "OFF ⌄ N@" to "F1", make the following actions:
● Enter ⟨ Y ⟩ ⟨ENTER⟩ .
● Then, enter ⟨ O ⟩ ⟨ F ⟩ ⟨ F ⟩ ⟨SPACE⟩ ⟨ N ⟩ ⟨SHIFT⟩ ⟨≡⟩ .

```
C H A N G E   F 1 =   ?  ( Y / N ) Y   O F F   N @ _
```

Now, push ⟨ENTER⟩ when the above message is put on the display. Then the computer will ask the following question.

```
C H A N G E   F 1 = O F F   N @   ?  ( Y / N )  _
```

If the entry is correct, observe the following:
● Return to the setup request after depression of ⟨ N ⟩ ⟨ENTER⟩ keys, or,
● Advance to F2 after pushing ⟨↓⟩ or ⟨ENTER⟩ key.

If the entry is not correct, reenter it after depressing ⟨ Y ⟩ ⟨ENTER⟩ .

NOTE: "@" will be changed to the CR code before output.
        Use of ⟨↓⟩ or ⟨ENTER⟩ key will enable changes and to check contents of F2, F3 ... of MODE I.

⟨↓⟩

```
C H A N G E   F 2 = & 0 A   ?  ( Y / N )  _
```

It defines "&0A=10 (LF code)" to F2, not "F2="&0A".
"0" thru "F" is usable for two digits after "&", otherwise, it will be handled as though nothing has defined.

⟨↓⟩

```
C H A N G E   F 3 =   ?  ( Y / N )  _
```

Nothing has been defined to "F".

⟨↓⟩

```
E   F = 4   J A M E S   B R O W N E   ?  ( Y / N )  _
```

Above is an example that the name "JAMES BROWNE" is stored.
If the description were to be too long, it hides the leading part of the data (i.e. CHANGE...).

⟨↓⟩

```
C H A N G E   F 5 =   G O S U B   ?  ( Y / N )  _
```

Above is an example that the subroutine jump statement "GOSUB" is implemented to F5 (MODE I) in the BASIC program mode.
Be careful of the fact that it outputs the internal code of "GOSUB" (2 bytes) in the terminal program mode, not the output of the ASCII code string "GOSUB".

54

[↓|

```
CHANGE  F6  ?  (Y / N) _
```

F6 is slightly different from F1 – F5. An equal symbol "=" will not be displayed after "F6". So, it becomes impossible to see what was defined to F6. Confirmation will also not be possible even in the BASIC program mode in direct. Though "F6" is convenient to keep information confidential like pass word that not to be known by any other party, its confidentiality will not be solely dependent.
To store "SOLAR", for example, first push   Y   ENTER .

```
CHANGE  F6  ?  (Y / N) Y _
```

Then, enter [s] [o] [L] [A] [R].

```
CHANGE  F6  ?  (Y / N) Y   SOLAR_
```

Now, enter ENTER .

```
CHANGE  F6  ?  (Y / N) _
```

If no change is to be made, push [↓] ENTER or N ENTER . Then, it will reutrn to the setup request.
Information in F1 ~ F6 are stored in the reserve area.
When there is no more room to define with F1 ~ F6 keys due to full occupancy of the reserve area, it will return to the setup request screen after putting the setting error message on the display.
The contents of "F1" thru "F6" will be cleared when the reserve program is newed or loaded in the BASIC mode.

3)   ↑

Used to check the length of the buffer that is used in the terminal program mode or to check and change communication parameters.
Depression of ↑ key, for instance, will put the following information on the display.

```
BUFFER  LENGTH = 5888  ?  (Y / N) _
```

It indicates that the memory size reserved in the terminal program mode is 5888 bytes, which will be obtained after the execution of "NEW 0 ENTER " with the 8 K RAM in the computer (1850 + 4096 – 58).
Maximum capacity without the use of the option memory module is 1850 – 58 = 1792 bytes.
Upon entry to the terminal program mode, it permits use of an entire free area (area that is not used for program and variables). Since the capacity is indicated on the display, it will be known how many bytes were secured in the terminal program mode.
The size can be changed within a range of 512 ~ free area –58.
To make it 1024, for instance, perform the following operation.
● [N] [ENTER] .
● [1] [0] [2] [4] .

55

```
LENGTH=5888  ? (Y/N) N  1024_
```

`ENTER`

```
BUFFER LENGTH=1024  ? (Y/N) _
```

If it is correct, push `↓` `ENTER` or `Y` `ENTER` .

```
BAUD  RATE=300  ? (Y/N) _
```

If indicates that the present baud rate is 300.
Now, let us discuss how to make change of parameters "300,8,N,1" to "300,7,E,1".
Since there is no change required for the baud rate, proceed to a next parameter screen.
(Same effect by the depression of `ENTER` or `Y` `ENTER` key.)

`↓`

```
WORD  LENGTH = 8  ? (Y/N) _
```

Now, WL=8 must be changed to WL=7.

Enter `N` `ENTER` and `7` .

```
WORD  LENGTH = 8  ? (Y/N) N  7_
```

`ENTER`

```
WORD  LENGTH = 7  ? (Y/N) _
```

`↓`

```
PARITY=N  ? (Y/N) _
```

PR=N must be changed to PR=E.

`N` `ENTER`

`E`

```
PARITY=N  ? (Y/N) N  E_
```

`ENTER`

```
PARITY=E  ? (Y/N) _
```

`↓`

```
STOP  BITS=1  ? (Y/N) _
```

```
  ┌──┐
  │ ↓ │
  └──┘
 ┌─────────────────────────────────────────────────────┐
 │ S e t u p :           A u t   F n c   C o m           │
 └─────────────────────────────────────────────────────┘
```
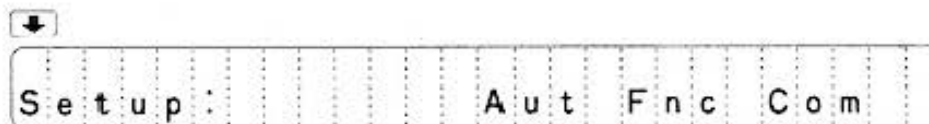
If any one of communication parameters were found incorrect, the control proceed to the setup request after putting the setting error message on the display.

[3] Operate

When ↓ is pushed while the setup request (c.f. (2)) is on the display, the following contents are brought on the display.

```
 ┌─────────────────────────────────────────────────────┐
 │ O p e r a t e :         N r m   A / P   A / L         │
 └─────────────────────────────────────────────────────┘
    |    ˙    |    ┌┐    |   ┌┐  |  ┌┐  |   ┌┐  |   ┌┐
```

It shows the automatic stop mode assignment in the terminal program mode.

1) ┌˙┐

When this key is pushed, the symbol "RUN" comes to activate.
It was set to this mode at the time of terminal program mode entry.
Data received through the RS-232C port and the key entry data during ECHO ON are stored in the buffer in the order they are received. When the buffer becomes fully occupied, the newest data going into the buffer pushes up the oldest data.

Present buffer contents

```
 ┌─────────────────────────────────────────────────────┐
 │ A B C D E - - - - - - - - - - - - - - - - - - - 0 1   │
 └─────────────────────────────────────────────────────┘
```

Input data = 2

```
 ┌─────────────────────────────────────────────────────┐
 │ A B C D E - - - - - - - - - - - - - - - - - 0 1 2     │
 └─────────────────────────────────────────────────────┘
```

Input data = 3

```
 ┌─────────────────────────────────────────────────────┐
 │ B C D E F - - - - - - - - - - - - - - - 0 1 2 3       │
 └─────────────────────────────────────────────────────┘
```

2) ┌˙┐

When this key is pushed, the symbol "PRO" comes to activate.
Though it has the almost similar action as (1), the data received through the RS-232C port or key input data during ECHO ON will be stopped receiving at every 512 bytes.
If the XON/XOFF mode is on, it will send out the X-OFF code (CONTROL S) so that no more data, except 30 bytes, will be received, if the transmitter should continue to send data.

3) ┌˙┐

When this key is pushed, the symbol "RESERVE" comes to activate.
Though it has the almost similar action as (2), the data received through the RS-232C port or key input data during ECHO ON will stop receiving each time the CR code is met.

When the receiving operation stops at (1) or (2), the symbol BUSY is activated. But, the contents of the key entry data can be sent out even while the BUSY symbol is on.

[4] Protocol

When 【↓】 is pushed in the condition of (3), the following contents will be brought on the display.

```
Protocol:   ·XO/O  Echo
  □    □    □    □    □    □
```

1) □

For example, depression of this key will bring the following contents on the display.

```
XON/XOFF  ON · ? (Y/N) _
```

This means that the XOFF code (CONTROL S) is sent when the CE-158 goes into the non-receiving state). Or, the XON code (CONTROL Q) is sent when it becomes ready to receive ∗).

So, enter;

【 N 】 【ENTER】

to perform any action other than resuning transmitting and stop with the XON and XOFF codes. After this, there will be no XON and XOFF codes issued even if the CE-158 goes into the non-receiving or receive ready state.
Now, it will return to the protocol display again.
Depression of 【↑】 at this point will bring the following prompt on the display.

```
XON/XOFF  OFF  ? (Y/N) _
```

Push 【↓】 【ENTER】 or 【Y】 【ENTER】 , if it is correct.
Then, it returns to the protocol display.

∗) : Non-receiving condition will be on when one of the conditions below is met.
(a) Depression of □ key (MODE III).
(b) During A/P or A/L mode of (3).
(c) Depression of a window key (【→】 , 【←】 , 【↑】 , 【↓】 , 【SHIFT】 【↑】 , 【SHIFT】 【↓】), except for the A/L and Trc modes.
(d) Too fast transmission of input data through the RS-232C port (display or parallel port output). Display will catch up with 300 bps.

The BUSY symbol will turn on for conditions (a) —
(c) Receiving condition will be on the next conditions.
(e) Entry of □ key (MODE I) for (a) — (c).
(f) Upon termination of the process for (d).

The BUSY symbol will go out.

NOTE-1: It will go into the terminal program mode with XON/XOFF in the ON state in the execution of the TERMINAL statement, or, with XON/XOFF in OFF state in the execution of the DTE statement.
NOTE-2: When the XOFF code is received during ON state of XON/XOFF, it will suspend the transmission until the XON code is received.

58

2) ⌐̇

This key determines whether the key input codes are to be put on the display (ECHO ON) or not (ECHO OFF).

It will go into the terminal program mode with ECHO OFF in the execution of the TERMINAL statement, or, with ECHO ON in the execution of the DTE statement.

When the key is depressed after the execution of the TERMINAL statement, the following prompt is brought on the display.

```
E C H O   O F F   ?   ( Y / N )  _
```

Enter [↑] [ENTER] or [Y] [ENTER] , if ECHO OFF is needed. Then, the display returns to the protocol screen.

If it should be ECHO ON, push the following keys:

[N] [ENTER] .

Then, the display returns to the protocol screen.

Second depression of the ⌐̇ key will bring the following assignment on the display.

```
E C H O   O N   ?   ( Y / N )  _
```

[5] Output

When [SHIFT] [↑] are pushed in (1) — (4), instead of ⌐̇ , ⌐̇ , ⌐̇ , the following contents are put on the display.

```
O u t p u t :   E x t   T r c   D s p   E t x
```

1) ⌐̇

This key determines whether the data received through the RS-232C port are to be printed on the CE-150 or the printer connected to the parallel port.

As the CE-150 has been assigned after the entry to the terminal program mode, it is in the EXT OFF state.

Depression of this key at that point will put the following prompt on the display.

```
E X T .   P R I N T E R   O F F   ?   ( Y / N )  _
```

To transfer the received data to the parallel port (EXT ON), push the following key sequence:

[N] [ENTER]

As it returns to the output screen, push ⌐̇ key again. Then, the following prompt will be put on the display.

```
E X T .   P R I N T E R   O N   ?   ( Y / N )
```

Push [↓] [ENTER] or [Y] [ENTER] , if it is correct.

Then, the display returns to the output screen.

2) ⌐⌐

This key determines whether the data received through the RS-232C port or key input data during ECHO ON are to be put on the display (TRACE OFF) or sent on the parallel port (TRACE ON)∗). (DIRECT SENDING OUT)

As it had been assigned to the TRACE OFF mode after the entry to the terminal program mode, depression of the key will put the following prompt on the display.

```
T R A C E   O F F   ?  ( Y / N )  _
```

To assign the TRACE ON mode, push the following key sequence:

    [N] [ENTER]

As it returns to the output screen, push ⌐⌐ key again. Then, the following prompt will be put on the display

```
T R A C E   O N   ?  ( Y / N )  _
```

Push [↓] [ENTER] or [Y] [ENTER] , if it is correct.
Then, the display returns to the output screen.
In case it is in the ETX OFF mode at this point, no output is carried out to the parallel port but is put on the display.

3) ⌐⌐

Data received from the RS-232C port is put on the display from left to right.

But, once a space or CR code was put on display column of 21st to 26th digit in the CLEAN TEXT ON mode (at the time of terminal program entry: WORD WRAP), it holds the present display screen as it is until a next word (space or CR code) is received or the data of 26 characters are received. Then, the present display is cleared and replaced with the next new word or the new data of 26 digits which will be put from the left of the display screen. Unless a space or CR code was received within 26 digits of the display, the display begins to shift from right to left.

```
T h i s   i s   J a m e s   _
```

Input data = Browne ␣

```
T h i s   i s   J a m e s   B r o w n e   _
```

Input data = speaking.

```
T h i s   i s   J a m e s   B r o w n e   _
```

Input data = CR code

```
s p e a k i n g .   _
```

If the input data was received incomplete, that contents will not be put on the display. Because data shift from right to the left of the display, all of data may sometime not be read completely if they were received continuously.

Depression of ☐ key will put the following prompt on the display.

```
C L E A N   T E X T   O N   ?   ( Y / N )  _
```

To turn into the CLEAN TEXT OFF mode (horizontal scrolling), push the following key sequence:
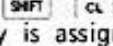
[N] [ENTER]

Then, it returns to the output screen.
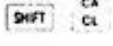
Depression of ☐ key again will put the following prompt on the display.
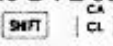
```
C L E A N   T E X T   O F F   ?   ( Y / N )  _
```

When no change is required, push [⬇] [ENTER] or [Y] [ENTER] . Then, it returns to the output screen.

4) ☐
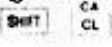
With this key is chosen assignment of EXT function to |cl| key and LONG SPACE PADDING to [SHFT] |cl| key.
The |cl| key is assigned to ETX function and [SHFT] |cl| keys to LONG SPACE after the execution of the DTE statement.
Both |cl| and [SHFT] |cl| causes no function after the execution of the TERMINAL statement.
When this key is pushed after the execution of the DTE statement, the following prompt is put on the display.

```
C L = E T X   O N   ?   ( Y / N )  _
```

Depression of [cl] key during execution of the terminal program mode will output the ETX code (03H) on the RS-232C port. Depression of [SHFT] |cl| key outputs LONG SPACE (continuous SPACE signals of 240 milliseconds).

To reset this function, push the following key sequence:

[N] [ENTER]

Then, it returns to the output screen.

Depression of the key again will put the following prompt on the display.

```
C L = E T X   O F F   ?   ( Y / N )  _
```

To maintain the present function, push [⬇] , [ENTER] or [Y] [ENTER] .

Menu display of [1] — [5] circulates in the following manner.
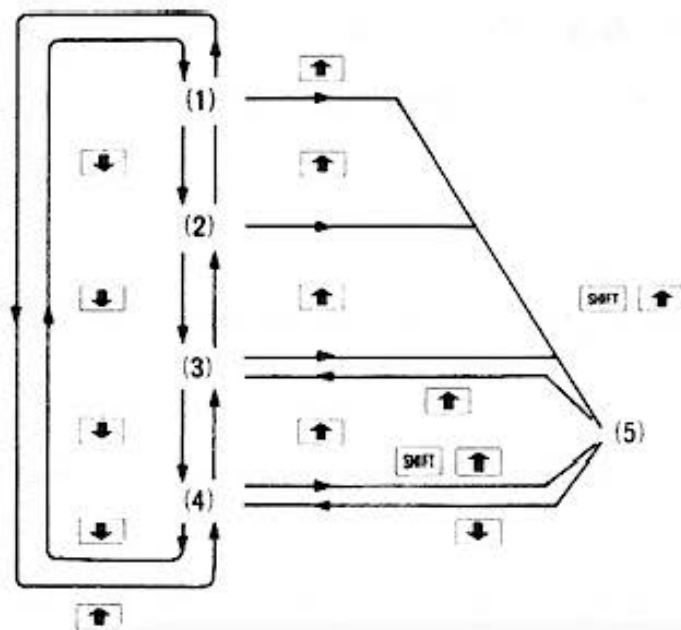
16. RTS, DTR signals

When it changes from the terminal display to the execution in the terminal program mode, both the RTS and DTR signals go into ON state.
It will return to the MENU SELECT mode when `ON` key is activated or LOW BATTERY condition is met, but both the RTS and DTR signals will be subject to the OUTSTAT statement executed before the terminal program mode.

17. Error codes

Listed below are error codes pertinent to the CE-158.

Error code	Description
50	1. Power switch of the CE-150 is off. 2. Needs to recharge NiCd battery.
51	1. Improper communication parameter designation by the SETCOM statement. 2. Improper communication device assignment by the SETDEV statement. 3. Less than 570 bytes of user area is reserved for the free area in entering into the terminal program mode with the TERMINAL statement of DTE statement.
52	Error found in the received data (parity, overrun error).
53	Improper TAB designation by the LPRINT statement. For example, the specified value exceeds the data size before the END code specified by the CONSOLE statement.
(54 ~ 57)	Not used.
58	Contents of the received data is not appropriate.
(59 ~ 60)	Not used.
61	Unmatched headers met during the execution of the CLOAD, CLOADr, MERGE, or INPUT# statement.
65	The number of variables in INPUT is less than that of Data Received thru RS-232 PORT.
67	1. No line number existing at the top of the line in executing the CLOADa or MERGEa statement. 2. For the CLOADa or MERGEa statement, the size of the receive data exceeded 160 or 80 in converting into the internal code. 3. The size of data before the CR code exceeded 80 in executing the INPUT or INPUT$ statement.

PARALLEL INTERFACE

PARALLEL INTERFACE

Parallel port

Specifications

Output format	:	Parallel (8 bits)
Transmission method	:	Handshake (ON with BUSY signal)
Code used	:	ASCII
Applicable statements	:	LPRINT, LLIST, FEED, CONSOLE, PRINT#--9,
Interface	:	TTL level

Parallel interface

(1) Specifications

Synchronization	:	By means of externally supplied strobe pulse.
Handshake	:	By BUSY signal.
Logic level	:	Output data and all interface control signals are TTL compatible.

— Electrical conditions on parallel interface connector pin —

[Output signal level]
HIGH = +2.4 to +5.0 V
LOW = ±0 to +0.4 V

[Output condition]
Outputted through 74LS04.
One TTL or LSTTL compatible is recommended for the load.

[Input signal level]
HIGH = +2.0 V to +5 V
LOW = ± 0 to +0.8 V

[Input condition]
Input load of one 74LS04.
Input signal is pulled up by means of 10 Kohms resistance.

(2) Output connector

DB-25(M) in use.
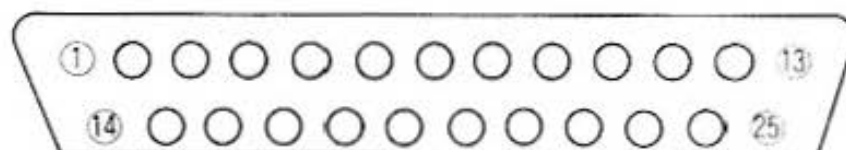
Connector pin assignment

The CE332P option interfacing cable (25 to 36 pin), EA-158C, is available.

(3) Connector signal descriptions

Pin No.	Signal name	-In/Out	Description
1	STROBE	Out	Data output strobe. Nromally high, but turns low during data output.
2	DATA 1	Out	
3	DATA 2	Out	
4	DATA 3	Out	8-bit data signal. High when the data is "1". Bit state is retained unitl output of next data after sending out the STROBE signal.
5	DATA 4	Out	
6	DATA 5	Out	
7	DATA 6	Out	
8	DATA 7	Out	
9	DATA 8	Out	
10	BUSY	In	When high, data output is held up assuming the peripheral is BUSY.
11	$\overline{\text{INIT}}$	Out	Normally high, but low state of signal is issued after power on. If BUSY for more than 10 seconds, it establishes ERROR 69.
12	N.C.		Non-Connection
13	N.C.		Non-Connection
14	GND		
15	GND		
16	GND		
17	GND		
18	GND		
19	GND		In logically GND level, and also is a twist pair pattern signal for each signal.
20	GND		
21	GND		
22	GND		
23	GND		
24	GND		
25	GND		

(4) Timing chart
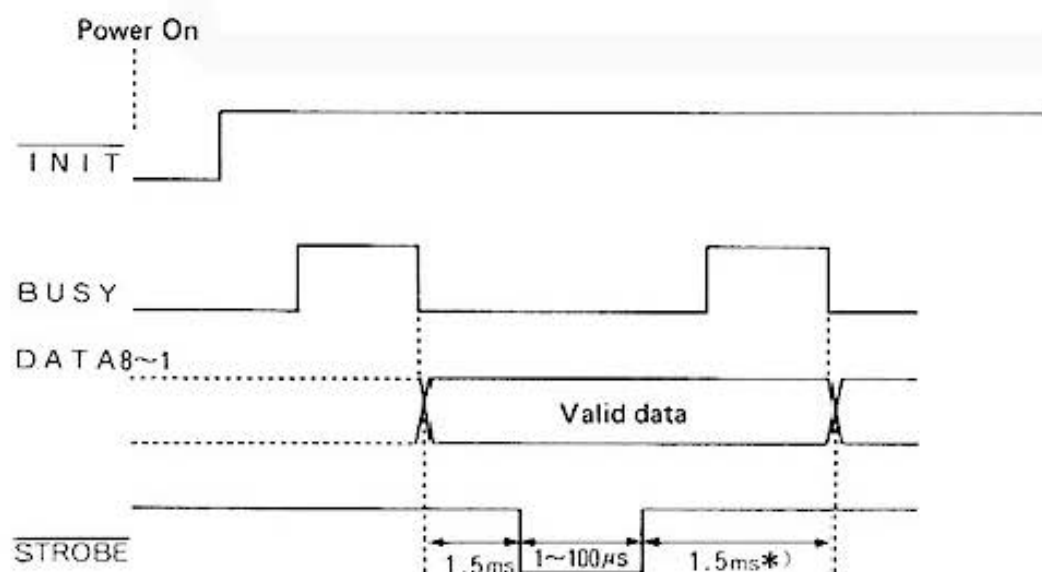
*): DATA 8 ~ 1 are retained for a minimum of 1.5 ms after rising of STROBE signal and further retained until the next DATA 8 ~ 1 is outputted (until BUSY turns low).

66

1. **Devices connectable**
   Handshaking by BUSY signal is adopted for the parallel interface of the CE-158.
   It becomes possible to make connection with the printer (CE-332P) that incorporates the Centronics parallel interface when the 25 pin to 36 pin conversion cable, EA-158C, is used.

2. **Cautions before connection**
   It will be a primary requisite to check the specification of the peripheral unit to be connected. Basic cautions will be described below.

   (1) Electrical characteristics of input/output pins
       Input/output pin of this unit is TTL compatible.

   (2) I/O signal timing
       The timing chart is shown in the separate page.
       Check if the peripheral unit to be connected meets with this timing.

   (3) Others
       If the printer is to be connected as a pheripheral device, check the carriage return and linefeed specification of the printer.
       Though the LF code is issued upon power on for the END code, it can be changed to the CR code, LF code, or a combination of both, using the CONSOLE statement. (Refer to "CONSOLE statement".)

   (4) When connected with the printer incorporating a normal parallel interface, it will not print " $\sqrt{}$ " and " π " on the printout as the normal type printer does not handle " $\sqrt{}$ " and " π ".
       However, in a course of program listing, " $\sqrt{}$ " will be printed as "SQR" and " π " as "PI".

   (5) Power ON/OFF of peripheral unit
       Power-on: Turn the PC-1500 on first and turn the unit on.
       Power-off: Turn the unit off first and turn the PC-1500 off.

3. **Interfacing cable**
   If the CE-158 is to be connected with the parallel interface printer other than the CE332P, check the electrical characteristics and signal flow of the input signal if they should meet with the signal characteristics of this unit.

4. Statements

Such as LPRINT, LLIST, FEED, CONSOLE, and PRINT#-9, are used for command statement for the parallel port.

With the LPRINT, LLIST, FEED, and CONSOLE statements,

(1) Command is executed to the parallel port after OPEN 'LPRT', or,

(2) not to the parallel port after OPN or SETDEV, but to CE-150 (LPRINT, LLIST) or RS-232C port (FEED, CONSOLE).

Statement		Manual execute			Program execute	Description
	Example	RUN	PRO	RESERVE		
OPN	OPN"LPRT"	O	O	O	O	LPRINT, LLIST, FEED, and CONSOLE statement after this will be executed to the parallel port.
	OPN	O	O	O	O	LPRINT and LLIST after this executed to the CE-150, and, FEED and CONSOLE after this is executed to the RS-232C port.
CONSOLE	CONSOLE 80,1	O	O	O	O	Sets digits/line and END code.
LPRINT	LPRINT 123	O	O	O	O	Data sent on parallel port.
LLIST	LLIST 10,100	O	O	O	O	Program sent on parallel port.
FEED	FEED 5	O	O	O	O	END code sent on parallel port.
PRINT#-9,	PRINT#-9, 123	O	O	O	O	Same as LPRINT.

(1) OPN

With this command is assigned the I/O port for the I/O statement.

All assignment by SETDEV are cleared upon execution of this command.

1) OPN 'LPRT'

The parallel port is assigned for the device.

LPRINT, LLIST, FEED, and CONSOLE statements after this are executed to the parallel port.

2) OPN

Cancels the device assignment, then the following operation is assumed.

- LPRINT and LLIST statements are executed to the CE-150. If the CE-150 was not connected, it results in ERROR 27.
- FEED and CONSOLE statements are executed to the RS-232C port.

Table in the next page described the port executed by the I/O statement of SETDEV and OPN.

Device set condition / Statement	A (1) OFF → ON (2) SETDEV (3) OPN	B — SETDEV					C OPN 'LPRT'
		KI	DO	PO	CI	CO	
(1) INPUT	Key input	Input from RS-232C port	Key input	←	←	←	←
(2) PRINT	Display output	←	Output to RS-232C port	Output to display	←	←	←
(3) LPRINT	Print on CE-150 (ERROR 27 if not connected)	←	←	Output to RS-232C port	Subject to A	←	Output to parallel port
(4) LLIST	Print on CE-150 (ERROR 27 if not connected)	←	←	Output to RS-232C port	Subject to A	←	Output to parallel port
(5) CLOAD	Load from cassette (ditto)	←	←	←	Input from RS-232C port	Subject to A	←
(6) MERGE	Load from cassette (ditto)	←	←	←	Input from RS-232C port	Subject to A	←
(7) INPUT#	Load from cassette (ERROR 35 if not connected)	←	←	←	Input from RS-232C port	Subject to A	←
(8) CSAVE	Save to cassette (ERROR 27 if not connected)	←	←	←	←	Output to RS-232C port	Subject to A
(9) PRINT#	Save to cassette (ERROR 35 if not connected)	←	←	←	←	Output to RS-232C port	Subject to A
(10) FEED	Output to RS-232C port	←	←	←	←	←	Output to parallel port
(11) CONSOLE	Format setup of RS-232C port	←	←	←	←	←	Format setup of RS-232C port
(12) TAB	Output to CE-150 (ERROR 27 if not connected)	ERROR 27	←	←	←	←	←

(2) CONSOLE
1) Specifies number of columns per line. That is, the END code is issued after sending to the parallel port the specified number of data.
2) Specifies the kind of the END code sent to the parallel port.

Upon power on, the item (1) has been set to 80 digits/line and the item (2) to the LF code.

Same syntax is applied as the RS-232C interface section, the specified columns/line and the END code are only effective when executed to the parallel port. It is not effective to the RS-232C port.

(3) LPRINT, LLIST, FEED
Same syntax is applied as the RS-232C interface section, but it differs in the following two points.
1) Executed to the parallel port.
2) Columns/line and the END code is subject to the description of the CONSOLE statement after the execution of OPN 'LPRT'.
Set to 80 digits/line and the LF code for the END code upon power on.

(4) PRINT#-9,
Same as LPRINT statement of (3), except that it executes to the parallel port regardless of OPN 'LPRT' declaration.

5. Initial state upon power on after [OFF] key power off
When power is turned on to the PC-1500, the parallel interface section is held in the following initial state.

Item	Initial state	Description
CONSOLE	80 ,1	80 digits/line, END code = LF code
TAB	0	

When power is turned on after auto power off, the above condition returns to the state before power off.
But, as the parallel interface connector signal $\overline{INIT}$ goes low temporarily during power on, it may cause the peripheral to reset, depending on the type of peripheral connected.

70

6. Error code

There is an ERROR 69 as the error code pertinent to the parallel port.
Details of other errors related to command execution are described in the RS-232C interface description.

Error No.	Description
69	If BUSY signal continues to be high for more than 10 seconds in issuing data on the parallel port. Cause for such a condition may be one of followings: ● Peripheral device not in connection ● Power is not turn on to the peripheral device ● Peripheral device in BUSY state ● Error in the peripheral device ● Paper empty in the printer operation In case it turns to BUSY state for more than 10 seconds even if the peripheral is in good working condition, adjust it by means of the software or handle it using "ON ERROR GOTO" and "ERL" statements.

POWER SUPPLY

This unit is powered by a rechargeable Ni-Cad battery. It may be necessary to recharge the battery the first time you use the CE-158.
If this unit has not been used for three months or more you will probably need to recharge the battery

This unit can also be operated on AC power through use of an AC adaptor EA-21A.
(Please note that the CE-158 can be used after about 5 minutes by connecting the AC adaptor EA-21A, when the voltage of the built-in rechargeable battery is low.)

Recharging:

When the CE-158 is connected with the Computer, or the CE-158 is connected with the Computer and the CE-150, the following message indicates that the battery should be recharged.

1. When the computer is turned on with the CE-158 power ON:

      : CHECK 8   Note: CHECK 8 message means that the battery of CE-158
                                             should be recharged. If this appears, connect the AC
      or   adaptor to the CE-158 for recharging.
                                             If the CHECK 8 is displayed with the Computer power
      NEW0? : CHECK 8   ON even after recharging, the CE-158 may be malfunc-
                                             tioning. Please contact an authorized SHARP service
                                             center.

2. During operation

      ERROR 50

3. When the terminal program is executed with the CE-158 power ON:

      LOW BATTERY

To recharge the battery, turn the Computer (and the Printer, CE-150) and CE-158 power OFF, connect the AC adaptor (EA-21A) to the Interface, and plug the AC adaptor into a wall outlet. (See the diagram.)
It will take about 15 hours before the battery is fully charged.

How to connect the AC adaptor

Power switch

AC adaptor connection jack
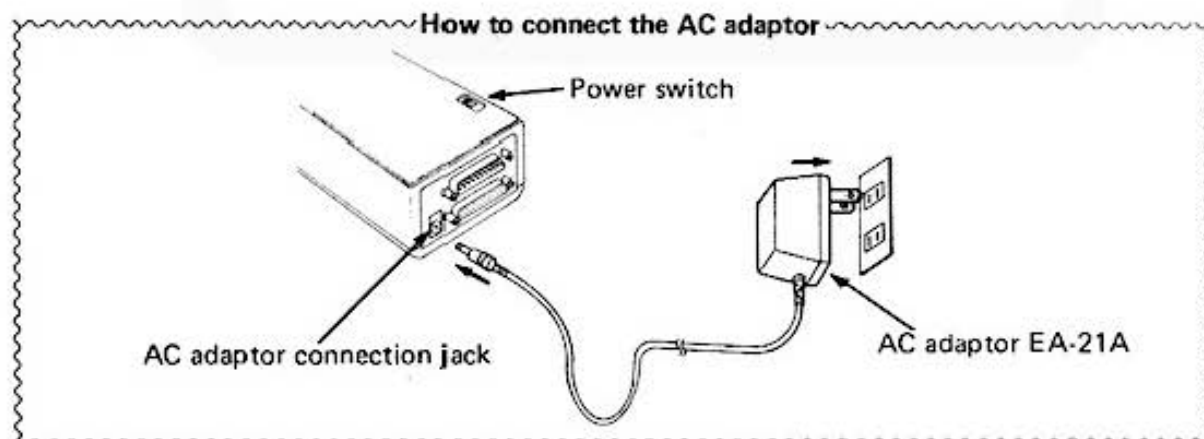
AC adaptor EA-21A

Important Note!

● Using any AC adaptor other than the one supplied (EA-21A) may damage the Interface.

● This unit can be operated about 3 hours when the battery is fully recharged.

● This unit can be recharged even when the CE-158 is disconnected with the Computer.

Notes:
- If you do not use your interface for a long time, the battery may lose power. If this happens, a 15-hour recharge will not be long enough to fully charge the battery.
  A slightly longer recharging is needed to make the interface fully charged.

- When the Interface has been left uncharged for an extended period of time after the battery had been exhausted, the battery power will be dropped beyond the minimum allowable charge level. In such a case, refresh the batteries to full capacity through recharging.

- When the unit is not in use, be sure to remove the adaptor from the interface and the wall outlet.

- Be sure to turn off power before connecting or disconnecting the adaptor.
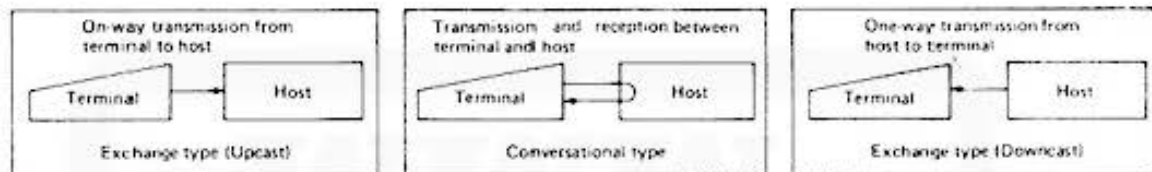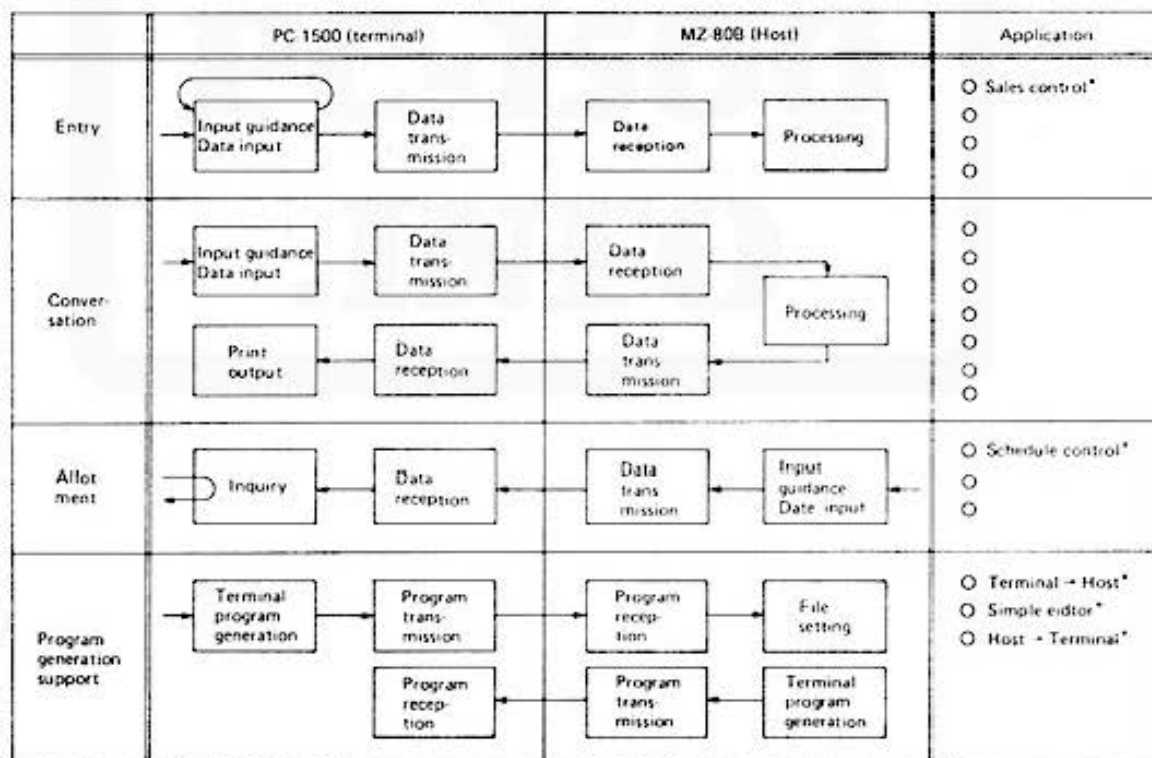
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~M E M O~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

PROGRAM EXAMPLES

When connected to each other, computers produce a number of advantages. The merits in utilizing informations include mutual utilization capability of program or data exchanges and executions, biased load elimination and one data base shared by several computers. In handling the computers, problems also include the capability of operation and preparation in remote places. When the 2-way power supply, compact and lightweight (pocketable) PC-1500 with a sufficient capacity is assumed as a terminal computer, data entry or program generation is possible from where you are, e.g., on a trip, outdoors or in bed. It is also possible for those who are limited to the use of a so-called host computer to enter data or generate programs in a free period of time. Thus, this results in a great convenience.

The following describes the application equipment and program examples with MZ-80B regarded as a host computer:

■ Use patterns of the CE-158 [RS-232C] interface

■ Application

With the applications marked with ＊ picked, the following describes their program examples. When using these programs in the PROGRAM EXAMPLES, the RS-232C interface of MZ-80B must be adjusted at modem mode and at 1200 bps in baud rate. Refer to the RS-232C instruction manual of MZ-80B for more detail.

75

| PROGRAM TITLE | SALES CONTROL | 1 |

Sales information input to the PC-1500 by individual is collectively transmitted to the MZ-80B as a host computer. (Exchange type · upcast transmission)

(1) Processing concept at terminal (PC-1500)
Input sales information by individual to the PC-1500.
The input information by individual includes 5 items of date, personnel in charge, product, unit price and quantity.
The PC-1500 is capable of information inputs by individual up to 10. Upon completion of the input, all the input data is transmitted to the MZ-80B by the transmission instruction. After sending it out the input area of the transmitter becomes empty, allowing inputs of another 10 kinds of information. After the power supply of the terminals is turned off by the BREAK operation, another terminal can send the data to the host.

(2) Processing concept at the host (MZ-80B)
Upon receipt of information by individual, the MZ-80B counts it as the number of received data, and at the same time, calculates sales per individual information. If required, the sales together with the received data are sorted into those by personnel in charge and by product for data printouts by a printer. After the printouts, the MZ-80B will be on standby for data reception.

■ **Data input/output formats**

(1) Data input and output formats (PC-1500)

(2) Data reception format (MZ-80B)

76

■ Example

	(1st time)			
Date	P.C	Product	Unit Price	Q'ty
9 /29	AOKI	PC-1500	59.800	2
9 /30	HIGUCH	CE-150	49.800	3
9 /30	OKAMOTO	PC-1500	59.800	1
10/ 1	OKAMOTO	CE-150	49.800	1
10/ 3	AOKI	PC-1500	59.800	2
10/21	HIGUCHI	MZ-80B	278.000	1

	(2nd time)			
Date	P.C	Product	Unit Price	Q'ty
10/25	HIKAGE	MZ-2000	148.000	7
10/25	AOKI	CE-150	49.800	4
10/27	HIGUCHI	PC-1500	59.800	3
10/28	OKAMOTO	PC-1500	59.800	10
10/28	OKAMOTO	CE-150	49.800	10
10/30	HIKAGE	MZ-2000	148.000	5

Sales results listed on the above are input for transmission by separating them in two times.
The number of inputs at a time is up to 10 kinds of information.

■ Printouts

```
          * SALES REPORT *            NO. 1

DATE    PERSONNEL IN CHARGE    PRODUCT          UNIT PRICE  Q'TY      AMOUNT

929       AOKI                 PC-1500              59800     2       119600
930       HIGUCHI              CE-150               49800     3       149400
930       OKAMOTO              PC-1500              59800     1        59800
1001      OKAMOTO              CE-150               49800     1        49800
1003      AOKI                 PC-1500              59800     2       119600
1021      HIGUCHI              MZ-80B              278000     1       278000


          * SALES CONTROL LIST *

DATE    PERSONNEL IN CHARGE    PRODUCT          UNIT PRICE  Q'TY      AMOUNT

1003      AOKI                 PC-1500              59800     2       119600
929       AOKI                 PC-1500              59800     2       119600
930       HIGUCHI              CE-150               49800     3       149400
1021      HIGUCHI              MZ-80B              278000     1       278000
1001      OKAMOTO              CE-150               49800     1        49800
930       OKAMOTO              PC-1500              59800     1        59800
          * SALES REPORT *            NO. 2

DATE    PERSONNEL IN CHARGE    PRODUCT          UNIT PRICE  Q'TY      AMOUNT

1025      HIKAGE               MZ-2000             148000     7      1036000
1025      AOKI                 CE-150               49800     4       199200
1027      HIGUCHI              PC-1500              59800     3       179400
1028      OKAMOTO              PC-1500              59800    10       598000
1028      OKAMOTO              CE-150               49800    10       498000
1030      HIKAGE               MZ-2000             148000     5       740000


          * SALES CONTROL LIST *

DATE    PERSONNEL IN CHARGE    PRODUCT          UNIT PRICE  Q'TY      AMOUNT

1025      AOKI                 CE-150               49800     4       199200
1003      AOKI                 PC-1500              59800     2       119600
929       AOKI                 PC-1500              59800     2       119600
930       HIGUCHI              CE-150               49800     3       149400
1021      HIGUCHI              MZ-80B              278000     1       278000
1027      HIGUCHI              PC-1500              59800     3       179400
1030      HIKAGE               MZ-2000             148000     5       740000
1025      HIKAGE               MZ-2000             148000     7      1036000
1028      OKAMOTO              CE-150               49800    10       498000
1001      OKAMOTO              CE-150               49800     1        49800
1028      OKAMOTO              PC-1500              59800    10       598000
930       OKAMOTO              PC-1500              59800     1        59800
```

77

■ Operation procedure

Data is input according to the guidance displayed on the screen.
However, when transmitting the data already gatherd and stored in the PC-1500, the program start must be initiated by ⌈DEF⌉ ⌈A⌉ key operation to protect the memories.

■ Flowchart

[PC-1500] at the terminal   [MZ-80B] at the host

*1 text is a telegram per processing unit and represents all input data in this case.

*1 text transmission

1 text reception

(Including already received data)

■ Program list (PC-1500)

```
10:DIM HI(10),TA$
   (10),SI$(10),T
   A(10),SU(10)
20:A=0
30:"A":ON ERROR
   GOTO 600
31:SETDEV
35:Y$="": INPUT "D
   ATA INPUT?(+):
   OUTPUT?(-)";Y$
40:IF Y$="+"THEN
   100
50:IF Y$="-"THEN
   300
60:GOTO 30
100:IF A=10THEN 20
    0
110:A=A+1
120:INPUT "DATE ?
    MMDD";HI(A)
130:INPUT "PERSONN
    EL IN CHARGE ?
    ";TA$(A)
140:INPUT "PRODUCT
    S ?";SI$(A)
150:INPUT "UNIT PR
    ICE ?";TA(A)
160:INPUT "QTY ?";
    SU(A)
170:Z$="": INPUT "D
    ATA END ?   (Y
    ) ";Z$
180:IF Z$="Y"THEN
    30
190:GOTO 100
200:PRINT "DATA FU
    LL      HIT EN
    T"
210:GOTO 30
300:OUTSTAT 0:IF A
    =0THEN 30
301:WAIT 60
310:PRINT "IN TRAN
    SMISSION ":
    WAIT 0
320:SETCOM 1200,8,
    N,1
330:SETDEV PO:
    LPRINT "ABC"
335:SETDEV KI:
    INPUT B$:IF B$
    <>"Y"BEEP 1:
    GOTO 320
337:SETDEV PO
340:B=1
350:IF B>ATHEN 500
360:IF B<>ATHEN
    LPRINT " "
370:IF A=BTHEN
    LPRINT "E"
380:LPRINT USING "
    #####";HI(B)
390:LPRINT TA$(B)
400:LPRINT SI$(B)
410:LPRINT USING "
    #########";TA(B
    )
```

```
420:LPRINT USING "
    ####";SU(B)
421:SETDEV KI
422:INPUT X$
423:SETDEV PO
430:B=B+1
440:GOTO 350
500:SETDEV
520:FOR C=1TO A
530:HI(C)=0:TA$(C)
    =""
540:SI$(C)="":TA(C
    )=0
550:SU(C)=0
560:NEXT C
570:A=0
580:GOTO 30
600:IF STATUS 4=31
    THEN 35
610:STOP
```

STATUS 1

834

Memory contents

A	DATA (END11) POINT
B	DATA TRANSMISSION COUNT
C	DATA CLA COUNT
X$	RECEPTION OK RETURN
Y$	MENU SELECTION
Z$	DATA END?
HI(10)	DATE
TA$(10)	PERSONNEL IN CHARGE
SI$(10)	PRODUCT
TA(10)	UNIT PRICE
SU(10)	QUANTITY

80

■ Program List (MZ-80B)

```
90 CONSOLE C40
100 RSMODE A,RB,[8,M70,RX1
110 DIM HI(200),TA$(200),SI$(200),TA(200),SU(200)
120 DIM CH(200)
125 T=1:U=1:A=0:Z=0:B=1:NB=0:N=0
130 PRINT CHR$(6)
140 CURSOR 5,3
150 PRINT "** SALES CONTROL PROGRAM **"
210 B=N+1
214 ON ERROR GOTO 217
215 RSMODE A,RB,T8,M70,RX1
216 RSI A B$
217 FOR D=1TO 500:NEXT D
218 IF B$="ABC"THEN RSO A "Y":GOTO 220
219 RSO A "N":GOTO 214
220 RSI A B$
230 E$=B$
240 RSI A B$
250 HI(A+B)=VAL(B$)
260 RSI A B$
270 TA$(A+B)=LEFT$(B$+SPACE$(16),16)
280 RSI A B$
290 SI$(A+B)=LEFT$(B$+SPACE$(16),16)
300 RSI A B$
310 TA(A+B)=VAL(B$)
320 RSI A B$
330 SU(A+B)=VAL(B$)
340 RSO A "0"
350 IF E$="E" THEN 380
360 B=B+1
370 GOTO 220
380 IF A+N>200 THEN 900
400 NB=N+1:N=B:Z=Z+1
410 PRINT /P "          * SALES REPORT *          NO.";Z;"          ":PRINT/P
420 PRINT /P " DATE  PERSONNEL IN CHARGE     PRODUCT          UNIT PRICE Q TY
    AMOUNT"
421 PRINT/P
430 FOR B=NB TO N
440 PRINT /P HI(A+B) ,
450 PRINT /P TA$(A+B),
460 PRINT /P SI$(A+B),
470 PRINT /P TA(A+B) ,
480 PRINT /P SU(A+B) ,
485 PRINT /P TA(A+B)*SU(A+B)
490 NEXT B
520 CURSOR 5,15:INPUT "Is Sorting Required ? (Y) ";P$
521 IF P$="Y" THEN 550
523 GOTO 130
550 FOR B=1 TO N
570 J=T:K=A+B
580 IF (A=0)*(B=1) THEN T=K:GOTO 740
590 IF TA$(J)<>TA$(K) THEN 620
600 IF SI$(J)>=SI$(K) THEN 690
610 GOTO 630
620 IF TA$(J)>TA$(K) THEN 690
630 IF J=U THEN 660
640 L=J:J=CH(J)
650 GOTO 590
660 CH(J)=K
670 U=K
680 GOTO 740
690 IF J=T THEN 720
700 CH(K)=J:CH(L)=K
710 GOTO 740
720 T=K
730 CH(K)=J
740 NEXT B
741 A=A+N :N=0 :NB=0
764 P$=""
775 PRINT/P:PRINT/P:PRINT/P:PRINT/P:PRINT/P:PRINT/P
780 PRINT /P "          * SALES CONTROL LIST *  ":PRINT/P
790 PRINT /P " DATE  PERSONNEL IN CHARGE     PRODUCT          UNIT PRICE Q TY
    AMOUNT"
795 PRINT/P
800 C=T
810 PRINT /P HI(C) ,
820 PRINT /P TA$(C),
830 PRINT /P SI$(C),
840 PRINT /P TA(C) ,
850 PRINT /P SU(C) ,
855 PRINT /P TA(C)*SU(C)
860 IF C=U THEN 890
870 C=CH(C)
880 GOTO 810
890 GOTO 130
900 PRINT CHR$(6):CURSOR 5,10:PRINT "  **** DATA FULL **** "
910 N=200-A
920 GOTO 400
```

81

Memory contents

A	DATA END POINT
B	TEMPORARY VARIABLE (for reception data)
C	TEMPORARY VARIABLE (for printout)
J	TEMPORARY VARIABLE (for sorting)
K	TEMPORARY VARIABLE (for sorting)
L	TEMPORARY VARIABLE (for sorting)
N	NO. OF RECEPTIONS
T	CHAIN ENTRANCE
U	CHAIN END
Z	NO. OF RECEPTIONS
B$	TEMPORARY RECEPTION BUFFER
E$	RECEPTION END FLAG
P$	OUTPUT AFTER SORTING?
HI(200)	DATE
TA$(200)	PERSONNEL IN CHARGE
SI$(200)	PRODUCT
TA(200)	UNIT PRICE
SV(200)	QUANTITY
CH(200)	DATA CHAIN

82

| PROGRAM TITLE | SCHEDULE CONTROL | 2 |

The scheduled data randomly input to the MZ-80B is selected by designating the date for transmission to the PC-1500. The number of the data to be transmitted is up to 20 kinds from the designated date. (Exchange type · downcast data transmission)

(1) Schedule input (MZ-80B)
The date and schedule contents are input according to the guidance. The date represents the day, hour and minute with no months distinguished.

(2) Data transmission to the PC-1500
The date is designated at the MZ-80B to instruct data transmission.

(3) Schedule inquiry (PC-1500)
The receiver (PC-1500) of the schedule data is capable of schedule inquiries even when the CE-158 RS-232C interface is disconnected from the PC-1500. The old or new schedule contents for the date are sequentially displayed by using the ↓ or ↑ key.

■ Data input/output formats
(1) Data input format (MZ-80B)

○ Each schedule input (Numerals) (Characters) Input | Date | Schedule

(2) Data sort format (MZ-80B)
○ Data sort
Date (number) data is sorted in the ascending order.
Example: The smallest data
Next smaller data
Next smaller data
The largest data

○ Transmission data designation
Example:
First transmission data (date)

(3) Data reception format (PC-1500)
| Date | Schedule

Reception

Transmission

83

Do not sell this PDF !!!

■ **Example**

Date		Schedule
9th	9:00	Morning gathering
9th	15:00	Mr. Sharp visits our company.
9th	18:00	Meets Mr. Hayakawa at Palace hotel.
9th	18:29	Hikari No. 165 at Tokyo station.
10th	8:30	Visits Sharp Corporation.
10th	10:00	Conference
10th	15:30	Hikari No. 180 at Shin-Osaka station.

• Input the schedule listed on the left and sort it in the order of old dates.

At the time of transmission, designate the first transmission data. The number of data to be transmitted at a time is up to 20.

■ **Printouts**

```
    90900:
Morning Gathering
    91000:
Meeting
    91500:
Mr.Sharp Vi.Our Co
    91800:
Meet Mr.Hayakawa P
    91824:
TOKYO st H-165
    100830:
Visit SHARP CORP.
    101534:
ShinOSAKA st H-180
```

```
 90900 : Morning Gathering
 91000 : Meeting
 91500 : Mr.Sharp Vi.Our Co
 91800 : Meet Mr.Hayakawa P
 91824 : TOKYO st H-165
100830 : Visit SHARP CORP.
101534 : ShinOSAKA st H-180
```

■ Operation procedure
Data is input according to the guidance displayed on the screen.

■ Flow Chart

[PC-1500] at the terminal

[MZ-80B] at the host computer

■ **Program List (PC-1500)**

```
 10:DIM TI(20),ME$
    (20)*18
 20:SETCOM 1200,8,
    N,1
 30:SETDEV
200:SETDEV KI
210:A=0
211:INPUT $B$:B=
    VAL (B$)
212:IF B=-1THEN 30
    0
220:INPUT $C$
223:INPUT $ME$(A)
228:TI(A)=VAL (C$)
231:SETDEV PO
232:LPRINT "O"
233:SETDEV KI
234:IF B=1THEN 300
240:A=A+1
250:IF A>20THEN
    STOP
260:GOTO 211
300:C=A
310:SETDEV
320:GOTO 400
400:"A":Z$="":
    INPUT "DISP(=)
    :PRIN,INT(+)EX
    T(-)";Z$
410:IF Z$="="THEN
    500
420:IF Z$="+"THEN
    700
421:IF Z$="-"THEN
    800
430:GOTO 400
500:WAIT 1
510:A=0
520:PRINT USING "#
    ######";TI(A);
    ":";ME$(A)
530:A$=INKEY$
540:IF A$=CHR$ (10
    )THEN 600
550:IF A$=CHR$ (11
    )THEN 650
560:IF A$=CHR$ (24
    )THEN 400
570:GOTO 530
600:A=A+1
610:IF A>CLET A=0
620:GOTO 520
650:A=A-1
660:IF A<0LET A=C
670:GOTO 520
700:CSIZE 2:LPRINT
730:FOR A=0TO C
740:LPRINT USING "
    #######";TI(A)
    ;":":LPRINT ME
    $(A)
750:NEXT A
770:GOTO 400
800:CONSOLE 80,1
810:OPN "LPRT"
820:FOR A=0TO C
```

```
830:LPRINT USING "
    #######";TI(A)
    ;" : ";ME$(A)
840:NEXT A
850:OPN
860:GOTO 400
```

STATUS 1

682

Memory contents

A	DATA POINT
B	END ELAG
C	END POINT
A$	CONTROLER FOR UPWARD AND DOWNWARD SHIFT
B$	RECEPTION BUFFER
C$	RECEPTION BUFFER
Z$	SELECTION OF DISPIAY MEDIA
TI(20)	DATE
ME$(20) *(20)	MESSAGE

■ Program List (MZ-80B)

```
120 DIM TI(200),MES$(200),CH(200)
140 RSMODE A,RB,1B,M70,RX1
170 T=1:U=1:V=0
190 CONSOLE C40
210 PRINT CHR$(6)
220 CURSOR 3,5
230 PRINT "*** SCHEDULE CONTROL PROGRAM ***"
240 CURSOR 5,7
250 PRINT "SCHEDULE ENTRY ? ---------- (1)"
260 CURSOR 5,9
280 PRINT "SCHEDULE TRANSMISSION ? ---- (2)"
290 CURSOR 5,15
300 INPUT Y$
310 Y=VAL(Y$)
320 IF Y=1 THEN 401
330 IF Y=2 THEN 1010
340 GOTO 290
401 A=V
410 PRINT CHR$(6)
420 CURSOR 5,5
430 PRINT "** SCHEDULE ENTRY **"
450 A=A+1
490 CURSOR 5,10
500 INPUT " DATE & TIME (DDHHMM) ";B$
510 TI(A)=VAL(B$)
511 IF TI(A)=0 THEN GOSUB 1400:GOTO 490
512 CURSOR 0,9:PRINT SPACE$(39)
520 CURSOR 5,12
530 PRINT " SCHEDULE CONTENTS "
531 CURSOR 5,14:INPUT "";B$
535 MES$(A)=LEFT$(B$+SPACE$(18),18)
560 CURSOR 0,16
570 INPUT "DATA END ? (Y/N)";A$
580 IF A$="Y" THEN 610
590 GOTO 410
610 FOR M=V+1 TO A
620 J=1:K=M
630 IF M=1 THEN T=M:GOTO 780
650 IF TI(J)<=TI(K) THEN 730
670 IF J=U THEN 700
680 L=J:J=CH(J)
690 GOTO 650
700 CH(J)=K
710 U=K
720 GOTO 780
730 IF J=T THEN 760
740 CH(L)=J:CH(L)=K
750 GOTO 780
760 T=K
770 CH(K)=J
780 NEXT M
790 V=A
800 GOTO 210
1010 PRINT CHR$(6)
1020 CURSOR 5,5
1030 PRINT "** SCHEDULE TRANSMISSION **"
1040 CURSOR 5,10
1050 PRINT "FIRST TRANSMISSION DATA ? (DDHHMM)"
1051 CURSOR 5,12
1052 INPUT "";P
1060 PRINT CHR$(6)
1070 CURSOR 5,10
1075 PRINT " ** IN TRANSMISSION **"
1080 REM
1100 B=T
1110 IF TI(B)>=P THEN 1190
1120 IF B=U THEN 1300
1130 B=CH(B)
1140 GOTO 1110
1190 C=0
1200 C=C+1
1210 IF C>20 THEN 210
1220 IF (C=20) + (B=U) THEN RSO A STR$(1)
1230 IF (C<20) * (B<>U) THEN RSO A STR$(0)
1241 FOR H=1 TO 1000:NEXT H
1250 RSO A STR$(TI(B))
1251 FOR H=1 TO 1000:NEXT H
1260 RSO A MES$(B)
1261 RSI A X$
1270 IF B=U THEN 210
1271 FOR H=1 TO 1000:NEXT H
1280 B=CH(B)
1290 GOTO 1200
1300 RSO A STR$(-1)
1310 GOTO 210
1400 CURSOR 0,0:PRINT "INPUT NUMERAL !"
1410 RETURN
1501 CURSOR 5,12
```

Memory contents

A	FOR INPUT
B	FOR TRANSMITION
C	COUNT OF TRANSMISSION
H	SYNCHRONUS VARIABLE
J	FOR SORTING
K	FOR SORTING
L	FOR SORTING
M	FOR SORTING
P	FIRST DATE
T	START DATA
U	END DATA
V	NUMBER OF DATA
Y	SELECTION IN MENU
A$	DISPLAY OF DATA END
B$	DATA INPUT BUFFER
X$	RECOGNITION OF RECEPTION
Y$	SELECTION IN MENU
TI(200)	DATE
MES$(200)	MESSAGE
CH(200)	CHAIN

87

| PROGRAM
TITLE | **PROGRAM GENERATION SUPPORT** | 3 |

Connection between the PC-1500 and the MZ-80B makes it possible to add the following new functions to the PC-1500 program generation method:
(1) The MZ-80B disc is utilized as the PC-1500 program file.
(2) The PC-1500 program is generated by using the screen of the MZ-80B.
(3) The PC-1500 program list is printed on a printer connected to the MZ-80B.

These functions can be optimized to make program generation more efficient.

[A] Program registration into the MZ-80B file

The program generated on the PC-1500 is registered into the MZ-80B disc file with a file name.

○ Operation procedure

(1) at MZ-80B

RUN [CR]

(file name) [CR]

(2) at PC-1500 (PRO-mode)

SETCOM 1200, 8, N, 1 [ENTER]

SETDEV CO [ENTER]

OUTSTAT 0 [ENTER]

CSAVEa [ENTER]

○ **Program list**

```
300 REM ---------------------------  SAVE PROCESS
310 CONSOLE C40
320 ON ERROR GOTO 450
330 RSMODE A,R8,T8,M70,RX1
340 INPUT "File name ?";F$
345 F$=LEFT$(F$+SPACE$(6),6)
350 WOPEN #1,FD1,F$
360 RSI A A$
370 IF A$="" THEN 410
380 PRINT #1 ,A$
390 PRINT      A$
400 GOTO 360
410 CLOSE #1
420 PRINT "----------- END --------- "
430 END
450 IF (ERL=350) * (ERN=44) THEN 470
460 STOP
470 PRINT " This File aleady exsists."
480 RESUME 330
```

Memory contents

A $	PROGRAM DATA
F $	FILE NAME

[B] Simple editor for the PC-1500 by using the MZ-80B

a) Program generation is possible on the CRT screen of MZ-80B.
   In generating a program, the following are possible:
   1) Line deletion and addition
   2) Character deletion and addition
b) Programs can be registered in the disc files.
c) Programs in the disc files can be loaded into the MZ-80B memory.
d) Program lists can be printed.
(Remark) MZ-80B is not allowed to execute the edited program to debug, therefore, make it after loading into PC-1500.

[CRT]

Position A: Program line No. (1 ~ 30000 and up to 200 lines) or COMMANDS input
Position B: Statement (up to 72 digits) or operand input
Position C: Program display by LIST command (up to 18 lines)
Position D: Error message display

[COMMANDS]

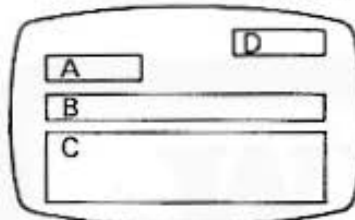|	Position A	Position B	Description
Program setting	Line number	Program statement	Program is set into the relevant program area by pressing the ENT key.
Line display	Line number	Space	The designated line is indicated on the display by pressing the ENT key.
Line deletion	DELETE	xxx [−xxx] line number line number	The designated line is deleted by pressing the ENT key.
Program list display	LIST	xxxx	20 lines from the designated line are indicated on the display by pressing the ENT key.
Program printout	LLIST	xxxx−xxxx	Lines in the designated range are printed out by pressing the ENT key.
Program registration into disc file	SAVE	xxxxxx 6 digits	Registration with the designated file name is made into the file by pressing the ENT key.
Program readout	LOAD	xxxxxx 6 digits	Program with the designated file name is loaded by pressing the ENT key.
File deletion	KILL	xxxxxx 6 digits	Program with the designated file name is deleted from the disc file by pressing the ENT key.
Edit end	END	Space	Edit end and return to BASIC made.

89

■ Program list

```
100 ON ERROR GOTO 5210
110 DIM N(200),M$(200),CH(200)
120 PRINT CHR$(6)
130 CONSOLE C80
140 CONSOLE S0,4
150 FOR A=1 TO 200
160 N(A)=-1
170 NEXT A
200 N=0:Y1=6:Y2=1:Y$=SPACE$(6):PRINT CHR$(6)
201 CURSOR 40,0:PRINT " << LINE NUMBER/CMMAND >> "
210 GOSUB 6000 :C$=Y$
290 IF Y$="      " THEN GOSUB 5010:GOTO 200
291 A=1
292 IF A>6 THEN 300
293 B=ASC(MID$(Y$,A,1))
294 IF (47<B) * (B<58) THEN A=A+1:GOTO 293
300 K=INT(VAL(LEFT$(Y$,A-1)))
301 IF K>30000 THEN GOSUB 5070:GOTO 200
302 IF K<0 THEN GOSUB 5020:GOTO 200
310 IF K=0 THEN 360
320 H=K:GOSUB 2400
330 IF J=0 THEN N=1:GOTO 350
340 M=J:Y$=M$(M):GOTO 370
350 GOSUB 2500:M=J:IF J=0 THEN GOSUB 5060:GOTO 200
360 Y$=SPACE$(72)
370 Y1=72:Y2=3
375 CURSOR 40,0:PRINT " << PROGRAM/OPERAND >> "
380 GOSUB 6000
381 IF C$="LIST  " THEN GOSUB 1400:GOTO 200
382 IF C$="LLIST " THEN GOSUB 1600:GOTO 200
383 IF C$="SAVE  " THEN GOSUB 1200:GOTO 200
384 IF C$="LOAD  " THEN GOSUB 1000:GOTO 200
385 IF C$="DELETE" THEN GOSUB 2000:GOTO 200
386 IF C$="KILL  " THEN GOSUB 2800:GOTO 200
387 IF C$="END   " THEN 1300
388 IF (LEFT$(C$,1)<>" ")*(K=0) THEN GOSUB 5010:GOTO 200
389 IF K=0 THEN GOSUB 5020:GOTO 200
390 IF Y$=SPACE$(72) THEN 430
391 N(M)=K:M$(M)=Y$
400 H=M:IF N=1 THEN GOSUB 2600:GOTO 200
410 GOTO 200
430 IF N=0 THEN H=K:GOSUB 2900
440 GOTO 200
1000 REM  ------------------------------- LOAD  PROCESS
1050 A=A+1:IF A>200 THEN GOSUB 5060:CLOSE #1:RETURN
1010 F$=LEFT$(Y$,6)
1020 ROPEN #1,FD1,F$
1030 A=0   :1=1
1040 IF EOF(#1) THEN 1110
1050 A=A+1:IF A>200 THEN GOSUB 5060:GOTO 200
1060 INPUT #1,A$
1070 GOSUB 1140 :IF E=1 THEN A=A-1:GOTO 1040
1080 N(A)=J:M$(A)=MID$(A$+SPACE$(72),B,72)
1090 CH(A)=A+1
1100 GOTO 1040
1110 U=A :CLOSE #1
1120 RETURN
1130 REM ----------------------------- LINE NUM GET
1140 B=1:D=LEN(A$):E=0
1141 IF B>D THEN E=1 :RETURN
1150 C=ASC(MID$(A$,B,1))
1160 IF ((47<C)*(C<58)) + ((C=32)*(B=1)) THEN B=B+1:GOTO 1141
1170 J=INT(VAL(MID$(A$,1,B-1)))
1180 RETURN
1200 REM  ----------------------------- SAVE  PROCESS
1210 F$=LEFT$(Y$,6)
1220 WOPEN #1,FD1,F$
```

```
1230 A=1: IF A=0 THEN 1280
1240 PRINT #1,STR$(N(A))+M$(A)
1250 IF A=U THEN 1280
1260 A=CH(A)
1270 GOTO 1240
1280 CLOSE #1
1290 RETURN
1300 REM ------------------------------ END    PROCESS
1310 CONSOLE S0,24
1320 PRINT CHR$(6)
1330 END
1400 REM ------------------------------ LIST  PROCESS
1420 A=1
1421 IF A>6 THEN 1429
1422 B=ASC(MID$(Y$,A,1))
1423 IF (47<B) * (B<58) THEN A=A+1:GOTO 1421
1429 C=INT(VAL(MID$(Y$,1,A-1)))
1430 A=I :B=1
1435 IF U=0 THEN 1530
1440 IF N(A)>=C THEN 1460
1441 IF A=U THEN 1520
1450 A=CH(A):GOTO 1440
1460 CONSOLE S5,24
1470 PRINT CHR$(6)
1480 PRINT STR$(N(A))+M$(A)
1490 B=B+1:IF (B=19)+(A=U) THEN 1510
1500 A=CH(A):GOTO 1480
1510 CONSOLE S0,4
1520 RETURN
1530 CONSOLE S5,24
1540 PRINT CHR$(6)
1550 GOTO 1510
1600 REM ------------------------------ LLIST PROCESS
1601 E=0
1605 IF U=0 THEN RETURN
1610 IF Y$=SPACE$(72) THEN C=0:D=999999:GOTO 1650
1620 E=1:B=0
1621 IF E>10 THEN 1625
1622 X$=MID$(Y$,E,1)
1623 IF X$="-" THEN B=E:GOTO 1625
1624 E=E+1:GOTO 1621
1625 E=0 :IF B=0 THEN E=1:B=10
1630 C=INT(VAL(MID$(Y$,1,B-1))):D=INT(VAL(MID$(Y$,B+1,10)))
1640 IF D=0 THEN D=999999
1650 A=T
1660 IF N(A)>=C THEN 1690
1670 IF A=U THEN 1740
1680 A=CH(A):GOTO 1660
1690 IF E=1 THEN 1750
1700 IF N(A)>D THEN 1740
1710 PRINT /P STR$(N(A))+M$(A)
1720 IF A=U THEN 1740
1730 A=CH(A):GOTO 1700
1740 RETURN
1750 IF C=N(A) THEN PRINT /P STR$(N(A))+M$(A)
1760 RETURN
2000 REM ------------------------------ DELETE PROCESS
2010 IF Y$=SPACE$(72) THEN GOSUB 5020 :RETURN
2020 E=1:B=0
2030 IF E>11 THEN 2070
2040 X$=MID$(Y$,E,1)
2050 IF X$="-" THEN B=E:GOTO 2070
2060 E=E+1:GOTO 2030
2070 E=0 :IF B=0 THEN E=1:B=10
2075 C=INT(VAL(MID$(Y$,1,B-1))):D=INT(VAL(MID$(Y$,B+1,10)))
2080 IF E=1 THEN D=C
2090 IF D=0 THEN GOSUB 5020:RETURN
2100 A=1 :F=0
2110 IF N(A)=C THEN 2140
2120 IF A=U THEN GOSUB 5020 :RETURN
```

91

```
2130 F=A:A=CH(A):GOTO 2110
2140 E=A
2150 IF N(A)=D THEN 2200
2180 IF A=U THEN GOSUB 5020:RETURN
2190 A=CH(A):GOTO 2150
2200 N(E)=-1
2210 IF E=A THEN 2230
2220 E=CH(E) :GOTO 2200
2230 CH(F)=CH(A):IF A=U THEN U=F
2240 RETURN
2400 REM  ------------------------------------- LINE NUMBER SEARCH
2410 A=T
2420 IF N(A)=H THEN 2450
2430 IF A=U THEN J=0:RETURN
2440 A=CH(A):GOTO 2420
2450 J=A:RETURN
2500 REM  ------------------------------------- NULL LINE SEARCH
2510 A=1
2520 IF N(A)=-1 THEN J=A:RETURN
2530 IF A=200 THEN J=0:RETURN
2540 A=A+1:GOTO 2520
2600 REM  ------------------------------------- SORT PROCESS
2610 A=H :B=T :IF T=0 THEN T=A:U=A:RETURN
2620 IF N(A)<N(B) THEN 2670
2630 IF B=U THEN 2660
2640 C=B:B=CH(B)
2650 GOTO 2620
2660 U=A:CH(B)=A:RETURN
2670 IF B=T THEN 2690
2680 CH(C)=A:CH(A)=B:RETURN
2690 T=A:CH(A)=B
2700 RETURN
2800 REM  ------------------------------------- KILL   PROCESS
2810 F$=MID$(Y$,1,6)
2820 DELETE FD1 F$
2830 RETURN
2900 REM  ------------------------------------- CHAIN OUT PROCESS
2910 A=T
2920 IF N(A)=H THEN 2960
2930 IF A=U THEN STOP
2940 B=A:A=CH(A)
2950 GOTO 2920
2960 IF A=T THEN 3020
2970 IF A=U THEN 3000
2980 N(A)=-1:CH(B)=CH(A)
2990 RETURN
3000 U=B:N(A)=-1
3010 RETURN
3020 IF A=U THEN 3040
3030 T=CH(A):N(A)=-1:RETURN
3040 T=0:U=0:RETURN
5000 REM  ------------------------------------- ERROR PROCESS
5010 ER=0:CURSOR 40,0:PRINT "  * COMMAND ERROR *         ":GOTO 5100
5020 ER=0:CURSOR 40,0:PRINT "  * UNDEFINED LINE *        ":GOTO 5100
5030 ER=0:CURSOR 40,0:PRINT "  * FILE NOT FOUND *        ":GOTO 5100
5040 ER=0:CURSOR 40,0:PRINT "  * FILE EXSIST *           ":GOTO 5100
5050 ER=0:CURSOR 40,0:PRINT "  * PRINTER ERROR *         ":GOTO 5100
5060 ER=0:CURSOR 40,0:PRINT "  * BUFFER FULL *           ":GOTO 5100
5070 ER=0:CURSOR 40,0:PRINT "  * LARGE LINE NUMBER *     ":GOTO 5100
5090 ER=0:CURSOR 40,0:PRINT "  * ?  ERROR *              ":GOTO 5100
5100 X$="A":MUSIC X$
5110 GET X$:IF X$="" THEN 5110
5120 CURSOR 40,0:PRINT "                              "
5130 IF ER=0 THEN RETURN
5140 ER=0 :GOTO 200
5200 REM  ------------------------------------- ERROR FIND PROCESS
5210 ER=1
5220 IF ERN=40 THEN RESUME 5030
5230 IF ERN=42 THEN RESUME 5040
5240 IF (64<ERN)*(ERN<68) THEN RESUME 5050
```

92

```
5280 IF ERN=1  THEN STOP "ERROR IN THIS PROGRAM
5290 RESUME 5090
6000 REM ----------------------------------- KEY INPUT PROCESS
6010 CURSOR 1,Y2:PRINT Y$;
6020 X1=1
6030 CURSOR 1,Y2+1:PRINT SPACE$(X1-1 );"^";SPACE$(Y1-X1+1);
6031 CURSOR X1,Y2
6040 GET   X$:IF X$="" THEN 6040
6041 X0=ASC(X$)
6050 IF X0<30 THEN 6100
6060 Y$=MID$(Y$,1,X1-1)+X$+MID$(Y$,X1+1,Y1-X1+1)    :PRINT X$;
6070 X1=X1+1
6080 IF X1>Y1 THEN X1=Y1
6090 GOTO 6030
6100 IF X0=13 THEN 6250
6110 IF 8<X0 THEN 6030
6120 X0=X0+1:ON X0 GOTO 6030,6030,6030,6130,6150,6030,6170,6190,6220
6130 IF X1=Y1 THEN 6030
6140 X1=X1+1:GOTO 6030
6150 IF X1=1 THEN 6030
6160 X1=X1-1 :GOTO 6030
6170 Y$=SPACE$(Y1):X1=1:CURSOR 1,Y2:PRINT Y$;
6180 GOTO 6030
6190 IF X1=1 THEN 6030
6200 Y$=MID$(Y$,1,X1-2)+MID$(Y$,X1,Y1-X1+1)+" ":X1=X1-1
6210 CURSOR 1,Y2:PRINT Y$;:GOTO 6030
6220 IF X1=Y1 THEN 6030
6230 Y$=MID$(Y$,1,X1-1)+" "+MID$(Y$,X1,Y1-X1)
6240 CURSOR 1,Y2:PRINT Y$;:GOTO 6030
6250 CURSOR 1,Y2+1:PRINT SPACE$(Y1+1);
6260 RETURN
```
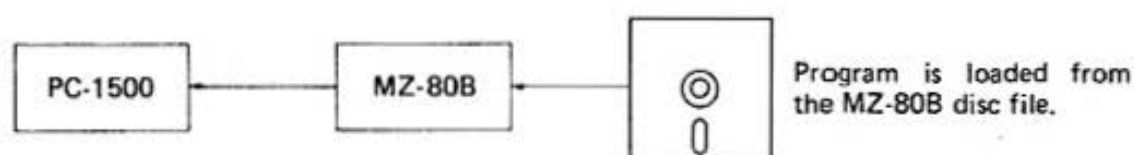
Memory contents

A	DATA INPUT ADDRESS		X1	NUMBER OF INPUT CHARACTER
B	√		X0	INPUT DATA CODE
C	√		Y1	ALLOWED NUMBER OF CHARACTERS
D	√		Y2	INPUT POINT(Y)
E	√		CH(200)	CHAIN
F	√		N(200)	DATA OF LINE NO
H	√		M$(200)	DATA OF PROGRAM STATEMENT
J	√			
K	√			
M	√			
N	√			
T	√			
U	DATA END ADDRESS			
A$	DATA READ IN			
C$	√			
F$	FILE NAME			
X$	KEY INPUT DATA			
Y$	INPUT DATA			
ER	FLAG FOR ERROR			

[C] Program load from the MZ-80B disc file.

Program is loaded from the MZ-80B disc file.

○ Operation procedure

(1) at PC-1500 (PRO-mode)

SETCOM 1200, 8, N, 1 [ENTER]
SETDEV CI [ENTER]
OUTSTAT 0 [ENTER]
CLOADa [ENTER]

(2) at MZ-80B

RUN [CR]
(file name) [CR]

○ Program list

```
500 REM      ------------------------ LOAD PROCESS
510 CONSOLE C40 :B$="A":TEMPO 7
520 RSMODE A,R8,T8,M70,RX0
530 ON ERROR GOTO 650
540 INPUT "File name ?";F$
545 F$=LEFT$(F$+SPACE$(6),6)
550 ROPEN #1,FD1,F$
560 IF EOF(#1) THEN 610
570 INPUT #1,A$
580 RSO A A$
581 MUSIC B$
590 FOR T=1 TO 2000:NEXT T
600 GOTO 560
610 RSO A CHR$(13)
620 CLOSE #1
630 PRINT "--------------- END -----------"
640 END
650 IF (ERL=550) * (ERN=40) THEN 670
660 STOP
670 PRINT "THis File does not exsist."
680 GOTO 540
```

Memory contents

A$	PROGRAM DATA
B$	DATA FOR SOUND
F$	FILE NAME

--M E M O---

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~**M E M O**~~~~~~~~~~~~~~~~~~~~~~~~~~~~

96

This apparatus complies with requirements of BS800 and EEC directive 76/889/EEC.

SHARP CORPORATION
OSAKA, JAPAN